Epigenomic and Genomic Analysis of Familial Prostate Cancer

By

Emma Cazaly, BMedRes (Hons)

Submitted in fulfilment of the requirements for the Degree of Doctor of Philosophy

University of Tasmania

September 2016

Declaration of Originality

This thesis contains no material accepted for a degree or diploma by the University or any other institution, except where duly acknowledged in the thesis and to the best of my knowledge and belief no material previously published or written by another person, except where due acknowledgment is made in the text of the thesis. Nor does the thesis contain any material that infringes copyright.

Emma Cazaly

Authority of Access Statement

This thesis may be made available for loan and limited copying and communication in accordance with the *Copyright Act of 1968.*

Emma Cazaly

Statement of Ethical Conduct

The research associated with this thesis abides by the international and Australian codes on human and animal experimentation, the guidelines by the Australian Government's Office of the Gene Technology Regulator and the rulings of the Safety, Ethics and Institutional Biosafety Committees of the University.

Emma Cazaly

Statement of Co-Authorship

Excerpts of the published review, (Cazaly *et al.* 2015) are incorporated in Chapter 1 of this thesis, with the review included in Appendix 1.1. This work was co-authored with Dr. Jac Charlesworth, Dr. Joanne L. Dickinson and Dr. Adele F. Holloway. The contribution of the authors to the publication are as follows: the primary research, laboratory and statistical analysis, text and figures were generated and prepared by EC. JC aided in drafting the manuscript. AFH and JLD participated in the conceptual design of the review and were substantially involved in drafting the manuscript.

Research contained within Chapter 3 of this thesis has been published as (Cazaly *et al.* 2016), included as Appendix 3.3. This work was co-authored with Dr. Russell Thomson, James R. Marthick, Dr. Adele F. Holloway, Dr. Jac Charlesworth, Dr. Joanne L. Dickinson. The contribution of the authors to the publication are as follows: the primary research, laboratory and statistical analysis, text and figures were generated and prepared by EC. RT participated in study design and provided direction for the statistical analysis. JM provided molecular laboratory support. AFH participated in study design and aided in drafting the manuscript. JC participated in study design, provided assistance with analysis and aided in drafting the manuscript. JLD participated in study design and was substantially involved in drafting the manuscript.

Emma Cazaly

Acknowledgements

Five years ago, as I came to the end of four months travelling and volunteering through the Indian subcontinent, I decided I wanted to yet again prolong my move from Tasmania and undertake my PhD in Hobart. However, there was only one project that interested me in the state. Fascinated by cancer epigenetics since first hearing of the field as an undergrad, I decided to email the lecturer who had inspired me all those years ago and see if by some miracle she had a project that I could start the following year. I will be ever grateful to Adele Holloway for responding to that hopeful email and together with Jo Dickinson taking a risk on me and creating such an exciting, challenging project. I am probably the most pleasantly astounded of us all to have "pulled it off". For all the ideas, support and hour upon hour spent reading my work, I am exceedingly grateful. Particular thanks to Adele for ticking off seemingly endless meetings and paperwork and always asking me the most important question – are you ok?

I'm especially thankful for my diverse supervisory team, with combined skills and knowledge to match the cross-disciplinary nature of my study. From Adele's extensive epigenetic knowledge, to Jo's prostate cancer genetics expertise, Jac's experience in familial studies and Russell's invaluable guidance in the statistical and analysis aspects of my study, thank you for all you have taught me.

A heartfelt thankyou to all the members of the Cancer, Genetics and Immunology group, particularly my office buddies in 502; your laughter, ideas, chats and friendship has made coming to work something I look forward to. Particular thanks to James Marthick for his endless patience and support in the lab, to Annette Banks for assistance with the familial resource- right to the end! And of course, to Nick Blackburn for all his bioinformatics support and many, many coffee and macaroon de-brief excursions.

To all the other amazing, fun-loving, inspiring people I have met through my candidature over the last four years- at MSP, other faculties and around the world, I am indebted to you for making this experience so much more than an

academic one. Particularly, Adro, Jimmy, Saner, Nas and Alice for the Monday morning coffees, Friday afternoon beers and everything in between. These are the moments that allow people to claim university days as being some of the best times of their lives.

To my family, you are the inspiration behind all the long hours and sacrifices I have willingly made over the past decade of undergraduate and postgraduate study- your belief and pride in me will ever push me to my best. And to Ry, you've helped me through the hardest parts of this degree with a grace I could never have imagined.

Finally, but certainly not least of all, I owe a huge thank you to the various funding bodies that made this study possible. To the Australian Government for my Australian Postgraduate Award Scholarship, to the Royal Hobart Hospital Cancer Auxiliary for an additional PhD scholarship which allowed me to quit part-time work and focus fully on my studies, and to the Menzies Institute for Medical Research and the Graduate Research office of the University of Tasmania for funding my travel to national and international conferences to share the work of my study.

Thank you to the participants of the Tasmanian Familial Prostate Cancer Study, without whom this study would not have been possible. This project was also supported by funding from the David Collins Leukemia Foundation, the Cancer Council Tasmania, Cancer Australia and the Australian Research Council.

Abstract

Over a million men world-wide are affected by prostate cancer, with the disease particularly prevalent in Australia, with more than 20,000 men diagnosed annually across the country. There remain significant clinical challenges in diagnosis and treatment. A family history is a major risk factor, indicating an underlying genetic component, yet the majority of inherited factors contributing to disease remain to be elucidated. Identifying this unaccounted heritable contribution will extend our understanding of prostate cancer development and progression, and has the potential to improve diagnosis and treatment.

It is becoming evident that high-risk genetic variants often occur in regulatory regions of genes, the primary sites of epigenetic regulation, therefore mapping epigenetic changes may shed some light on missing heritiability. Epigenetic marks are chemical modifications to DNA or it's associated proteins, that do not alter the genomic sequence, yet play a key role in regulating gene expression. DNA methylation, the most frequently studied epigenetic mark, is influenced by a range of intrinsic and external factors including diet, lifestyle and age. However, it is becoming increasingly apparent that genetic drivers may have the greatest influence on epigenetic patterns. While genetically driven epigenetic profiles contribute to natural phenotypic variation, such alterations may also underpin part of the unexplained inherited contribution to complex disease risk.

Large pedigrees with clusters of affected individuals can provide invaluable insight into complex diseases, affording reduced genetic complexity. As such, this study utilises the unique Tasmanian Familial Prostate Cancer Resource to

further understand the inherited drivers of epigenetic change that can predispose men to prostate cancer. Particularly, this study focuses on identifying genetic variants that may trigger DNA methylation changes in regulatory regions of the genome. Such variants have been termed methylation quantitative trait loci, or meQTLs, and can be examined in a similar manner to expression quantitative trait loci.

Clusters of affected men, representing dense aggregates of prostate cancer incidence often spanning up to five generation, were selected from four large families in the Tasmanian Familial Prostate Cancer Study. Samples were analysed for genotype and methylation profiles, initially using array based techniques which were then validated and extended with bisulphite sequencing.

Fundamental to analyses of methylome data is normalization and batch correction, to ensure unwanted technical bias is removed while maintaining the biological information of interest. While pre-processing methodologies were available for analysis of matched disease and control samples, the development of an optomised pre-processing pipeline for the analysis of famililal data was required. Specifically, this included testing a range of normalisation methods with qualitative and quantitative performance metrics to determine which method was the most appropriate and effective on familial data.

Potential meQTLs of interest were then identified through two distinct approaches. The first approach prioritised the most variable methylation sites between individuals, while the second approach examined methylation surrounding previously identified prostate cancer risk loci. To test the selected meQTLs, methylation data was combined with genotype using a generalized

linear model accounting for kinship. After adjusting for multiple testing error, significant associations were prioritised using the following criteria; proximity to prostate cancer relevant genes and the presence of key regulatory elements. Through bisulphite sequencing, prioritised meQTLs were initially validated, followed by finer mapping of the influence of meQTLs on surrounding methylation profiles. Additionally, unaffected controls were drawn from the Tasmanian Prostate Cancer Case Control Study to examine differential methylation patterns between affected and unaffected individuals, with the aim of identifying predisposing variants.

Using this approach an meQTL associated with the tumour suppressor gene CASZ1 was identified. This meQTL, located at 1p36.22, showed genetically driven methylation patterns at the SNP, which extended approximately 150bp either side, to two additional CpGs. Distinct differential methylation profiles were also observed between cancer and control groups for the *CASZ1* region. This meQTL provides an intriguing basis for further investigation as dysregulation of the gene has been associated with an aggressive prostate cancer phenotype.

Abbreviations and Acronyms

1KGP:	1000 Genomes project
AIHW:	Australian Institute of Health and Welfare
Bp:	Base pair
CGI:	CpG island
Cis:	Acting proximally (within 1 Mb)
CpG:	Cytosine-guanine dinucleotide pair
CpG open sea:	CpG sites located distantly to CpG islands
CpG Shelf:	Genomic region 2Kb either side of CpG shore
CpG Shore:	Genomic region 2Kb either side of CpG island
CpG-SNP:	Single nucleotide polymorphisms at the cytosine or guanine of a CpG pair
DRE:	Digital Rectal Exam
eQTL:	Expression Quantitative Trait Loci
FDA:	United States Food and Drug Administration
GWAS:	Genome-wide Association Study
Kb:	Kilobase (1,000 base pairs)
LD:	Linkage Disequilibrium
MAF:	Minor allele frequency in a given population
Mb:	Mega base (1,000,000 base pairs)
MDS:	Multi-dimensional scaling
meQTL:	Methylation Quantitative Trait Loci
miRNA:	Micro RNA
ncRNA:	Non-coding RNA

PCA:	Principal Component Analysis
PHI:	Prostate Health Index
PSA:	Prostate Specific Antigen
QTL:	Quantitative Trait Loci
RR:	Relative Risk
SBE:	Single base extension
SNP:	Single nucleotide polymorphism
Trans:	Acting distally (further than 1 Mb or on a different chromosome)
UTR:	Untranslated region
VMR:	Variably methylated region

Table of Contents

Chapter 2 – Features of the Tasmanian Familial Prostate Cancer Resource guiding study design

Chapter 3 – Development of appropriate normalisation strategies for the analysis of germline familial methylation data

Chapter.4 Identification and prioritisation of me-QTLs

Appendices

Chapter 1 - Introduction

1.1 Prostate Cancer Incidence and Mortality

Prostate cancer considerably impacts on the life expectancy and quality of life of millions of men and their families world-wide. Globally, prostate cancer is the second most common cancer in men with around 1.1 million men diagnosed in 2012 (Ferlay *et al.* 2015), as shown in Figure 1.1A, which depicts global incidence and mortality rates of prostate cancer relative to other common cancers. The disease affects a significant number of Australian men, with just over 20,000 men diagnosed in 2012 and 17,250 new cases estimated to have been diagnosed in 2015, comprising almost a quarter of all male cancer cases that year (AIHW, 2016). Figure 1.1B highlights Australia-specific incidence rates relative to other common cancers. A considerable number of men die from prostate cancer in Australia, with 3,112 deaths reported in 2013, accounting for 13% of all cancer deaths that year, making it the second deadliest cancer after lung cancer (AIHW, 2014), as depicted in Figure 1.1B, illustrating Australia-specific mortality rates relative to other common cancers.

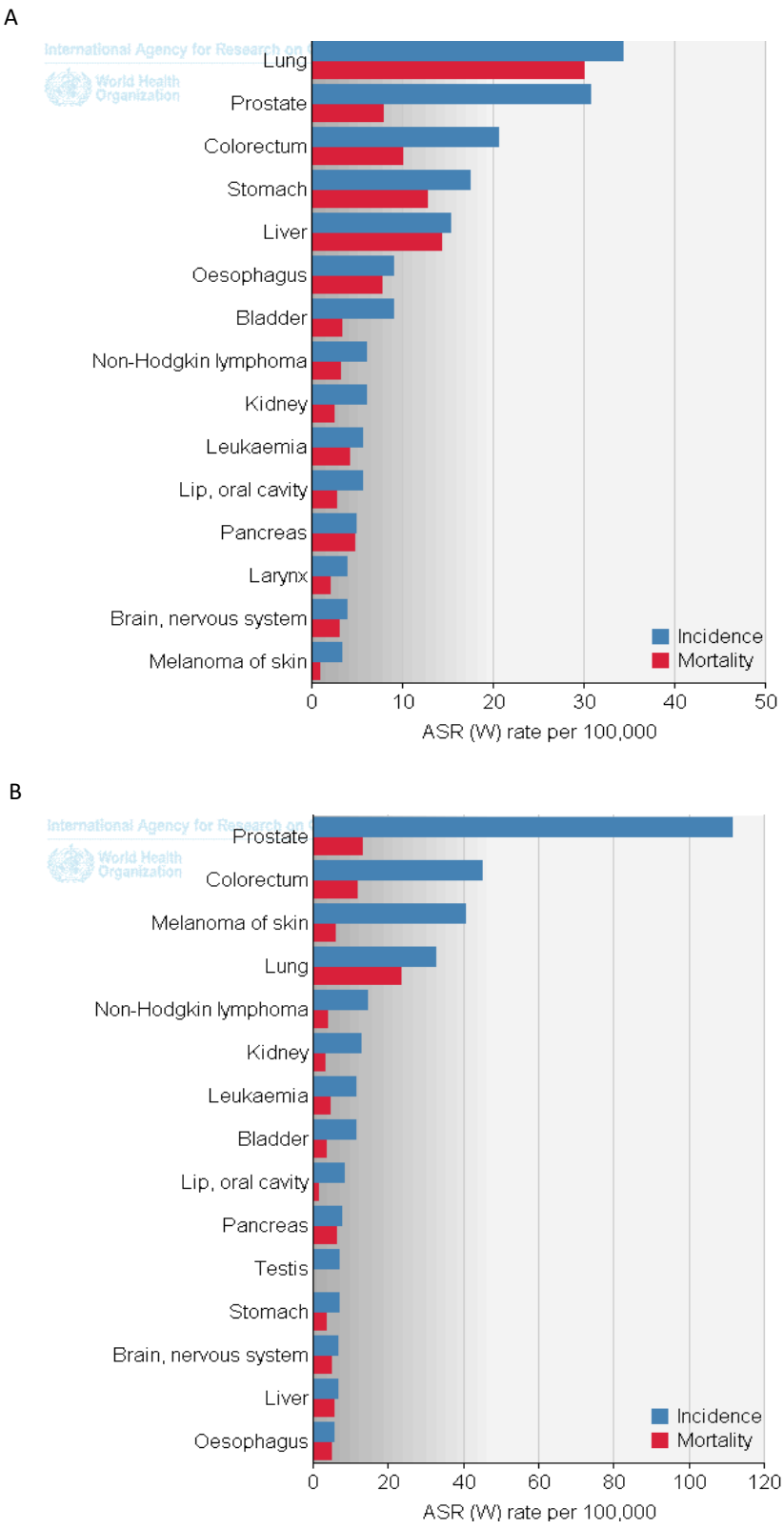


Figure 1.1 Incidence and mortality rates for various cancers in 2012.
Estimated age-standardised incidence and mortality rates for men across various cancer sub-types (A) globally and (B) specific to Australia and New Zealand. Adapted from the World Health Organisation's GLOBOCAN project, 2012. ASR (W): Age-standardised rates (world population)

Due to the widespread practise of prostate specific-antigen testing and subsequent biopsies, Australia and New Zealand have the highest reported incidence of prostate cancer worldwide, with 111.6 men per 100,000 diagnosed annually ((Ferlay *et al.* 2015), see Figure 1.1B). Prostate cancer is particularly pertinent for Tasmania men, with 446 men diagnosed in 2012, at an incidence of 174.6 per 100,000, compared to the Australia-wide incidence of 162.7 (111.6 as standardised to WHO world standard population) per 100,000 in the same year (AIHW, 2016). Tasmanian men also had a slightly higher mortality rate in 2012 at 28.2 per 100,000 compared to 27.8 Australia-wide.

1.2 Limitations of Prostate Cancer Diagnosis and Treatment

Diagnosis for prostate cancer is currently performed by digital rectal examination (DRE), serum levels of prostate-specific antigen (PSA) and prostate tissue biopsy. Tumours are graded using the Gleason score, measuring 1-10 depending on the abnormality of biopsied cells, however, for the majority of tumours with a mid-range score it is difficult to determine treatment course (DeWeerdt 2015). There is considerable debate regarding the utility of PSA testing, a blood test measuring levels of a protein normally produced by the prostate gland but often present in higher concentrations in men with prostate cancer (Sohn 2015). The test was originally developed in the mid 1980s to assess cancer progress post-treatment and was used in this manner until the mid 1990s when high mortality rates from prostate cancer necessitated an additional diagnostic to DRE. PSA was chosen as it was found to improve early detection by 78% (Catalona *et al.* 1994). Thus, from 1994 until

recently, the test has been used as the gold standard for diagnosis of new prostate cancer cases.

However, conflicting evidence surrounding effect or influence on mortality and the propensity for over diagnosis has led to much controversy surrounding the use of PSA as a diagnostic measure (Roobol 2015). Levels of the antigen can vary substantially between cancer-free men due to infection, inflammation, benign prostatic hyperplasia or age and have also been shown to decline in later stages of cancer (Sohn 2015). Thus as an initial diagnostic measure, the test has low specificity with substantial false positive rates at around 40-60%, and poor sensitivity, with over 30% false negatives reported (Brawer and Lange 1989). Evidence for the limitations of the PSA test is provided by the USA National Cancer Institute's figures; for every 1,000 men who's PSA was tested regularly over ten years, around 120 will return false-positive results with only 110 receiving an accurate cancer diagnosis, and of those correctly diagnosed five will die of the cancer despite screening and nearly half will develop complications from treatment. Only one death will be avoided (Sohn 2015).

Consequently, a watchful waiting approach is now more frequently chosen if the cancer is of a low grade, symptoms are not present or the man is of an age or health status where he is more likely to die from comorbidities. This approach is becoming increasingly common with only around 50% of men diagnosed with prostate cancer now being treated in the USA as opposed to 90% ten years ago (Sohn 2015). However, early treatment is life saving for men with aggressive cancer. It is thus

crucial to be able to distinguish between indolent and aggressive forms of prostate cancer, with much research continuing into developing biomarkers that can facilitate early and accurate diagnosis and identify those at high risk of metastasis.

A biomarker is a biological molecule found in body fluid or tissue that can objectively distinguish normal from abnormal biology or healthy and disease states. An ideal biomarker would be detectable in blood or urine, yet be solely expressed in the neoplastic prostate (DeWeerdt 2015). While the traditional PSA test is still the most widely used biomarker for prostate cancer, two other biomarkers are currently approved by the United States Food and Drug Administration (FDA). The Prostate Health Index (PHI) uses a combination of free and bound PSA and improves cancer detection in men with high Gleeson scores as well as improving prognostic accuracy at biopsy (Saini 2016). *Prostate Cancer Antigen 3* (*PCA3*) is a long non-coding RNA gene that is overexpressed in prostate tumours. The antigen test exhibits improved specificity and predictive value compared to PSA testing, yet has extremely variable sensitivity ranging between 20-90% and is thus preferentially used in conjunction with PSA and DRE (Saini 2016).

A plethora of additional biomarkers for both diagnostic and prognostic purposes have been commercially developed but are yet to be FDA-approved. The *Transmembrane Protease, serine 2-ETS-related gene (TMPRSS2-ERG)* gene fusion occurs in around 50% of prostate tumours, however as a diagnostic test it lacks sensitivity and specificity. This is at least partially a result of tumour heterogeneity and varying gene-fusion frequencies between populations. The utility of the marker

when used in isolation is limited, however in combination with PCA3 and PSA it has been shown to increase diagnosis and prognosis accuracy (Leyten *et al.* 2014). The *Phosphatase and Tensin Homolog (PTEN)* tumour-suppressor gene deletion is also used as a prognostic marker, often in combination with *TMPRSS2-ERG* to determine a patient's response to therapy, as the combination of these genomic aberrations is associated with poor patient outcome (Boström *et al.* 2015).

Many of the other diagnostic and prognostic tests being developed utilise panels of genomic or protein markers, see (Prensner *et al.* 2012; Cary and Cooperberg 2013; Saini 2016) for further detail. Unfortunately poor design of clinical trials and the long process of FDA-approval has hindered the availability of these tests in the clinic (Prensner *et al.* 2012). Despite this recent surge in the development of diagnostic and prognostic markers, these tests only examine some of the alterations underpinning prostate cancer and in order to continue to improve diagnostic and prognostic tools much still remains to be understood about the molecular drivers of prostate cancer.

A central driver for improving these diagnostic measures and enabling early and accurate diagnosis, is the fact survival rates dramatically diminish once prostate cancer has metastasised. Five-year survival rates for men with metastatic prostate cancer are only a third of the rate for those men diagnosed with a localised disease (Hodson 2015). For metastatic prostate cancer hormone therapy is the first line of treatment, with the aim of suppressing prostate-stimulating androgens, particularly testosterone. Such treatments may be effective for several years before the tumours

become resistant, by which stage treatment merely adds months to a patient's life (Savage 2015). Thus once prostate cancer has metastasised it is ultimately incurable. By understanding what drives prostate cancer to metastasise in some men and not others we may be able to further understand the mechanism by which cancers become resistant to chemotherapy and develop treatments that circumvent this.

Moreover, the ability to distinguish men with an indolent, slow-growing form of prostate cancer would reduce un-necessary treatment, as often men suffering this form of cancer will die from other causes (Patrikidou *et al.* 2014). This is highlighted in the high ten-year and fifteen-year survival rates for localised prostate cancer at 93% and 77% respectively (AIHW, 2012). Since prostate cancer treatments such as radio- or chemotherapy and prostatectomy can often lead to infection, incontinence, impotence or even death, minimising treatment to only those men who require it will help to ensure quality of life is preserved (Penson *et al.* 2005). Reducing over-treatment would also diminish the medical and financial burden on these men and on the health care system, allowing more resources to be directed to men with aggressive cancer so that these individuals have improved access to life-saving treatment.

1.3 Prostate Cancer Risk Factors

Risk factors for prostate cancer include environmental exposures, lifestyle, diet, age, ancestry and genetic and epigenetic predisposition. Environmental exposure to chemicals such as Vinclozolin, an agricultural antifungal that disrupts endocrine

function, and the hormone bisphenol A (BPA), commonly found in food and beverage packaging, have also been associated with prostate abnormalities and a heightened risk of prostate cancer respectively (Anway and Skinner 2008; Tarapore *et al.* 2014). Higher urinary BPA levels have been found in prostate cancer affected men, particularly those over 65, and animal studies indicate the hormone may induce abnormalities at the centrosome (Tarapore *et al.* 2014).

As with many complex diseases, chronic inflammation, obesity and physical inactivity are linked to a higher risk of prostate cancer, although the underlying pathophysiological mechanisms are yet to be fully understood (Koul *et al.* 2010; Rhee, Vela and Chung 2016). Similarly, the effect of diet on prostate cancer risk has been inconsistently reported. Consequently a recent consortium pooled 15 cohort studies of over 842,000 men including more than 52,000 prostate cancer cases, to further investigate possible associations (Wu *et al.* 2016). Examining the link between processed and un-processed red meat, poultry, seafood and egg intake, the study found only a modest positive correlation between red meat of any kind and advanced prostate cancer. Consumption of seafood was not found to have an effect while poultry was negatively associated with risk of advanced and fatal cancers and a higher consumption of eggs was linked to an increased risk of advanced and fatal cancers. Supporting this association, an earlier study in 2011 found that men who ate more than 2.5 eggs a week had an 81% increased risk of developing a lethal form of prostate cancer than those who ate less than 0.5 eggs per week (Richman *et al.* 2011). Conversely a diet high in folate and other nutrients involved in one-carbon metabolism such as methionine and vitamin B6 has been shown to be protective

against prostate cancer, especially high-grade clinically relevant forms (Shannon *et al.* 2009). Additionally, dietary interventions such as selenium supplementation, increased vegetable intake, ibuprofen and aspirin have been found to attenuate the growth of cancer in some tumour stages in a subset of men. While this study provides evidence that prostate cancer risk posed by genetic factors may be ameliorated in a some individuals by diet, there exists a very complex interaction between genetic risk and environmental factors, the details of which are still not well understood (Loeb *et al.* 2015). To fully appreciate the contribution and interaction of environmental and genetic risk factors, it may first be imperative to understand the underlying mechanistic link between the two – namely epigenetic regulation.

Ancestry also influences the risk of prostate cancer, with the highest incidence in affluent regions such as North America, Oceania and western and northern Europe and the highest mortality rates in less developed regions such as South America, sub-Saharan Africa and the Caribbean (Center *et al.* 2012). While part of this risk variation may be driven by environment, lifestyle or screening practises, USA and foreign born Chinese, Japanese, Vietnamese and Filipino men living in the USA all had higher unfavourable risk profiles at diagnosis than Non-Hispanic Whites in a recent study (Lichtensztajn *et al.* 2014). The heightened risk was also not associated with diagnosis at a later clinical stage and varied by ethnic group, indicating these men had biological differences predisposing them to a more severe disease progression (Lichtensztajn *et al.* 2014).

The disparity in incidence between races suggests genetic factors may be important contributors to risk. Numerous studies have consistently reported a family history to be a major risk factor for the disease (Carter *et al.* 1992; Keetch *et al.* 1995; Gronberg, Damber and Damber 1996; Ghadirian *et al.* 1997), with the association between family history and increased risk of prostate cancer found to be 1.45 (95% CI = 1.12-1.89) in a large consortium examining the risk of family history on several cancer types (Jacobs *et al.* 2010).

1.3.1 Genetic Risk Factors

With an estimated heritability of 60% prostate cancer is one of the most heritable cancers (Hjelmborg *et al.* 2014). Accordingly, the relative risk of prostate cancer is 2.48 if a first degree relative has been diagnosed (Kicinski, Vangronsveld and Nawrot 2011), with an increased risk if a man has an affected brother rather than father. The relative risk rises again if two or more first-degree relatives are affected (Relative Risk: 4.39, (Kicinski, Vangronsveld and Nawrot 2011)). Despite this well-established familial link, a comprehensive understanding of how genetic variation contributes to prostate cancer predisposition is yet to be elucidated.

Early familial studies using linkage analysis examined the segregation of disease variants within families, identifying risk associated loci in many genes including *RNASEL*, a putative tumour suppressor gene which regulates proliferation and apoptosis (Carpten *et al.* 2002) and *MSR1*, a macrophage scavenger receptor (Xu *et al.* 2002). However, the restricted genomic coverage of the available technology at the time limited the success of familial linkage studies and there was a persistent

failure in replicating results (Edwards *et al.* 2003). More recently, the large consortiums such as the International Consortium for Prostate Cancer Genetics (ICPCG) have utilised evolving technology to validate previous findings and discover new risk loci. Examining 1,233 pedigrees in 2010 the consortium replicated evidence for eleven previously identified regions and found risk associations at two novel regions (Christensen, Bonnie and George 2010). Two years later the ICPCG consortium validated eight prostate cancer risk loci previously identified through genome-wide association studies (GWAS) (Jin *et al.* 2012), and this year, having conducted GWAS analysis on over 5 million exon single nucleotide polymorphisms (SNPs) in 2511 unrelated familial prostate cancer cases and 1382 controls, the consortium validated a further six previously identified risk loci (Teerlink *et al.* 2016).

The success of combining next-generation genomic technology and familial studies to discover and validate prostate cancer risk loci has been demonstrated by the identification of *HOXB13*. Familial linkage studies had implicated a region on chromosome 17 (q21-22) as potentially containing a prostate-cancer risk gene (Lange *et al.* 2003). Further next generation sequencing studies examining the 200 genes in this region found a recurrent G84E mutation in *HOXB13* which segregated with disease and was significantly more common in men with familial prostate cancer than sporadic cases (carrier frequency of 3.1% compared to 0.6%) (Ewing *et al.* 2012). Additional studies confirmed the rare mutation (Breyer *et al.* 2012) and found it explained around 1% of the familial risk of prostate cancer in the UK (Kote-Jarai *et al.* 2015).

Population based GWAS, which examine differences in genotype between large numbers of unrelated affected and unaffected individuals, have identified over 100 prostate cancer risk loci (Eeles *et al.* 2009; 2013; Olama *et al.* 2014). However, as is the case for many complex diseases, together these common variants explain only a fraction of the population burden of disease. For prostate cancer, only 33% of the inherited risk of disease is explained by these loci across various populations (Olama *et al.* 2014). Many explanations have been proposed for the unexplained genetic component of complex disease susceptibility. For example, the impact of large deletions, inversions or copy number variants may not be detected in GWAS examining single nucleotide variations. Complex gene-gene and gene-environment interactions, overestimating heritability, poor modelling and statistical application could further create a mismatch between the expected heritability of prostate cancer and the heritability explained by known variants (Eichler *et al.* 2010).

There is also debate as to whether this "missing heritability" is due to many common variants of low penetrance acting in an additive manner or if rare variants of high penetrance, difficult to detect in GWAS, contribute significantly to disease predisposition. To help answer this question and comprehensively catalogue variation in the human genome, the 1000 genomes project (1KGP) was launched in 2008 and completed in 2015, mapping genetic variants down to a frequency of 1% (Auton *et al.* 2015). Numerous ongoing ventures have since sequenced hundreds of thousands of human genomes and exomes, examining natural variation and variants linked to common and rare diseases. Some of the most prominent include the 100,000 Genome Project (http://www.genomicsengland.co.uk/the-100000-

genomes-project) and the Exome Aggregation Consortium (http://exac.broadinstitute.org). Together these projects have shown the importance of rare variants in understanding complex disease aetiology.

However, the computational load of examining rare variants in these large studies is still immense. The advent of affordable next generation sequencing has allowed for a revival of familial studies, where pedigree structures and enrichment of rare variants allows transmission to be tracked down generations, providing greater discovery power, a much more manageable computational task to examine rare disease-linked variation. An early example of the power of familial studies can be seen in the discovery of the association between cardiovascular disease, hypercholesterolemia and LDL receptor mutations in familial studies (reviewed in (Endo 2010)). The resultant improvement in the understanding of the molecular biology underlying hypercholesterolemia lead to the development of statins, a drug therapy with huge benefit to the general population, not only those families affected by the rare mutations. Similarly, laboratory exploration of rare mutations discovered in familial prostate cancer studies could improve the molecular understanding of disease aetiology, improving diagnosis and treatment options for a wider population of men.

1.4 The Role of Non-coding Variants in disease

The debate between the common and rare variant hypotheses has traditionally focused on "functional" variation in coding regions of the genome. The key to explaining the remaining inherited component of disease burden may come from further understanding the role of "regulatory" variants; those variants outside gene-

coding regions yet involved in controlling gene expression. Evidence for the molecular importance of non-coding variation is indicated by the proportion of non-coding disease-associated SNPs identified in GWAS, with over 90% located outside of genes (Hindorff *et al.* 2009). At least some of these SNPs affect gene regulatory mechanisms, modifying gene expression by altering transcription factor binding or directing altered epigenetic profiles (Furey and Sethupathy 2013). Accordingly, it has been suggested that cancer is not only the result of the accumulation of genetic mutations with age but also a disruption in epigenetic reprogramming (Feinberg and Tycko 2004).

1.5 Epigenetics

1.5.1 Epigenetics at a Glance

Conrad Waddington first coined the term 'epigenetics' in the early 1940s to integrate the existence of two related phenomenon; that genetically identical cells possess the capacity to differentiate into tissue specific structures with correlated functions and that gene-environment interactions can affect phenotypes (reprinted in (Waddington 2012)). The term epigenetics has since come to refer to the environment surrounding the DNA, with a current working definition characterising an epigenetic trait as "a stably heritable phenotype resulting from changes in a chromosome without alterations in the DNA sequence" (Berger *et al.* 2009). This term is most often used in reference to the inheritance of traits to a daughter cell during mitosis, but there is evidence, although still controversial, of germ line

transmission of epigenetic traits between generations (Skinner 2011; Pembrey *et al.* 2014). For a timeline of key advances in the field of epigenetics see Figure 1.2.

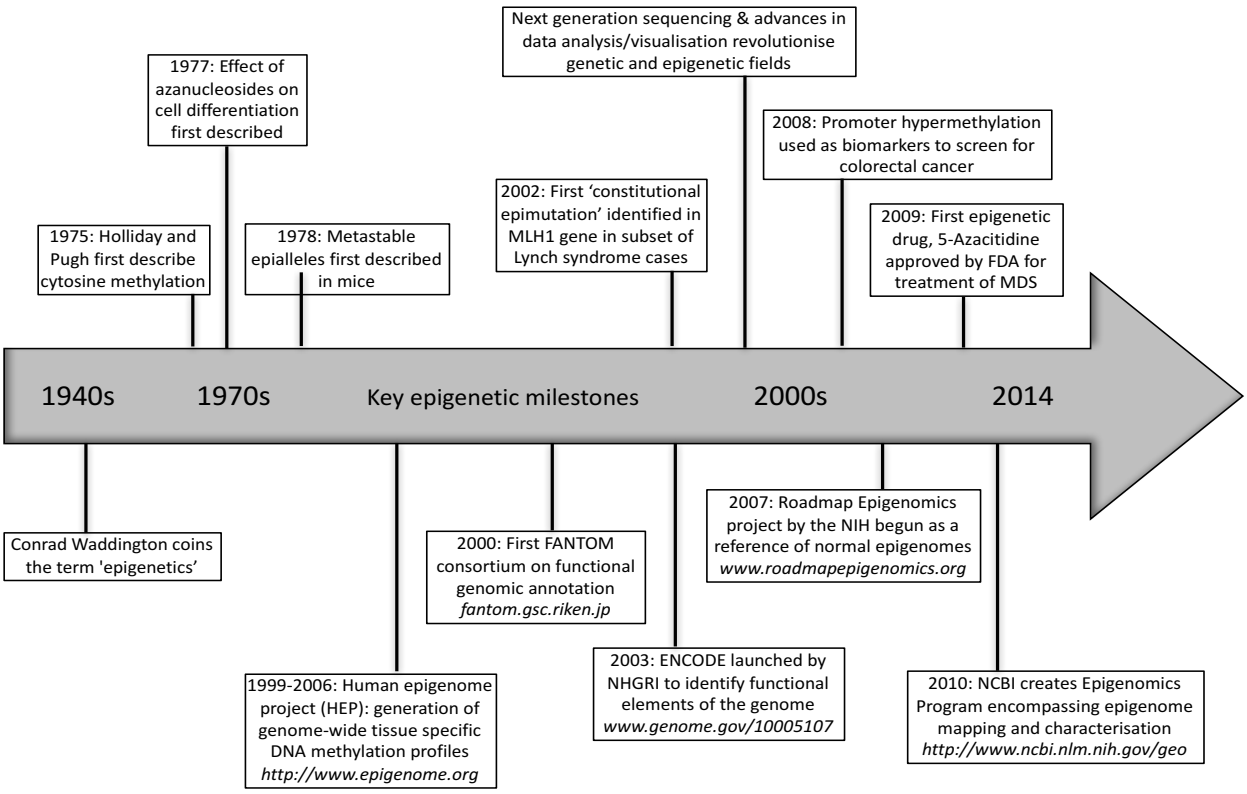


Figure 1.2 Timeline of key advances in the field of epigenetics.
Adapted from (Cazaly *et al.* 2015).

1.5.2 Epigenetic Modifications

Eukaryotic DNA is assembled into chromatin; repeating units of nucleosomes consisting of approximately 147 base pairs of DNA wrapped around an octamer of histones. Chromatin is then configured into higher order structures that play functional roles in regulating the accessibility of DNA to the transcriptional machinery. Mechanisms that influence the genomic environment and chromatin structure include modifications to the DNA itself, absence / presence of histone modifications and histone variants, and processes involving non-coding RNA and chromatin remodelling complexes (Dawson and Kouzarides 2012). At one extreme, tightly packaged, transcriptionally inactive heterochromatin is characterized by DNA methylation, de-acetylated histones and tightly packed nucleosomes, with non-coding RNA such as repressive miRNA involved in targeting and maintaining heterochromatin. At the other extreme, euchromatin contains unmethylated DNA, acetylated histones and active histone variants with DNA exposed to the transcription machinery (Richards and Elgin 2002). Epigenetic modifiers including epigenetic writers, readers and erasers, communicate with chromatin remodelling complexes to move and modify nucleosomes, opening or compacting chromatin. See Figure 1.3 for detail on the different epigenetic marks present in hetero- and euchromatin.

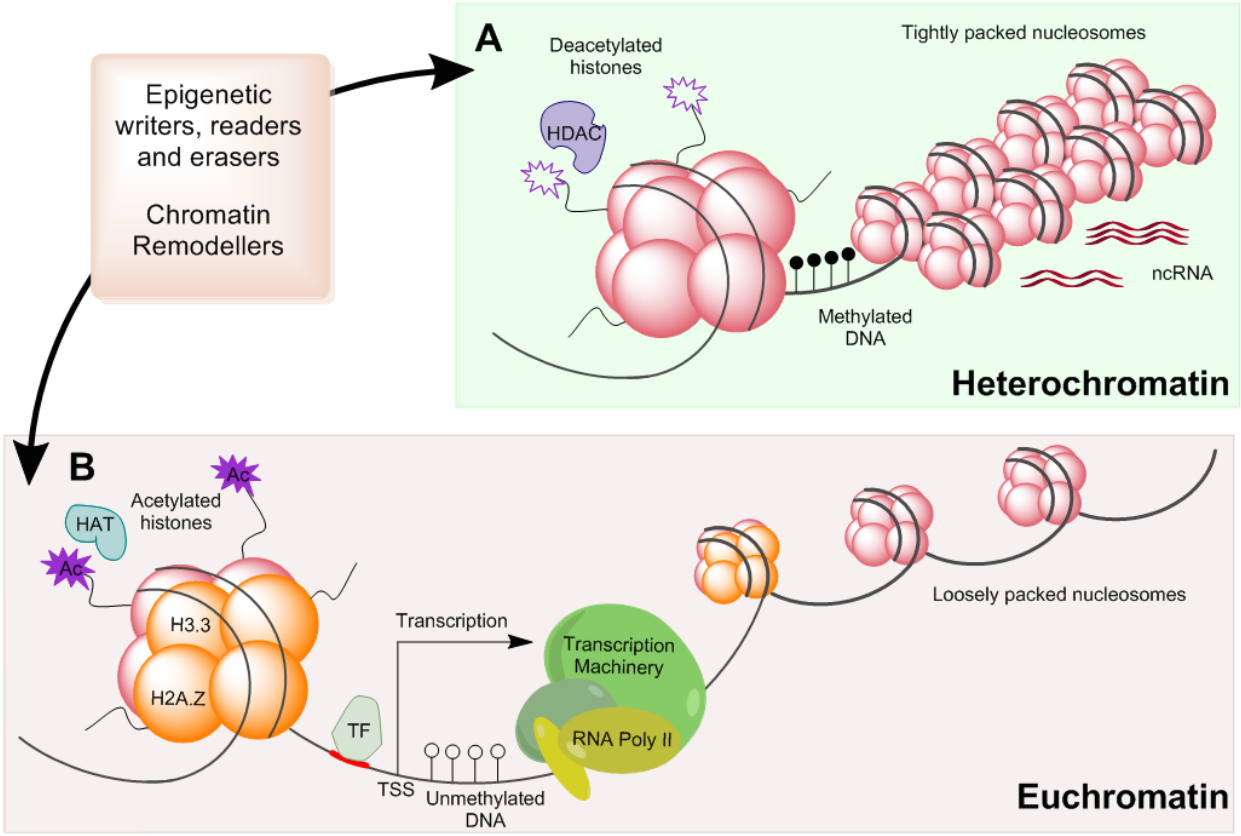


Figure 1.3 Epigenetic marks involved in chromatin regulation.
Higher order chromatin structures play a functional role in regulating the accessibility of DNA to transcriptional machinery. (**A**) Tightly packaged heterochromatin is characterized by DNA methylation (filled black circles), de-acetylated histones (clear purple stars) and tightly packed nucleosomes. (**B**) Alternatively, euchromatin contains unmethylated cytosines (unfilled circles) and acetylated histones (purple stars), with DNA exposed to the transcription machinery (green). Epigenetic modifiers including writers, readers and erasers, communicate with chromatin remodelling complexes to move and modify nucleosomes, opening or compacting chromatin. Adapted from (Cazaly *et al.* 2015).

More than a dozen post-translational modifications of histone proteins have been reported to date, including methylation, acetylation, phosphorylation and ubiquitination (Dawson and Kouzarides 2012), and technological advances have made it possible to map these modifications genome-wide (Barski *et al.* 2007; Consortium *et al.* 2015). These modifications have together been proposed to form a 'histone code' which can be interpreted by cellular proteins to specify downstream functions (Allis 2008). In addition, chromatin structure is altered by the actions of ATP-dependent chromatin remodelling enzymes and the exchange of canonical histones with histone variants. This creates a highly dynamic, adaptable epigenetic landscape that plays a key role in regulating genome function and provides an interface between the environment and the genome.

In recent years the existence of a complex network of non-coding RNAs (ncRNAs) transcribed from the human genome has become apparent. These ncRNAs have regulatory functions and play a key role in the establishment and maintenance of other epigenetic marks (Daxinger and Whitelaw 2012), with evidence that they constitute a mechanism for epigenetic inheritance through generations (Liebers, Rassoulzadegan and Lyko 2014). There is also mounting evidence for the involvement of ncRNAs in disease development, particularly in cancer (Kasinski and Slack 2011).

1.5.3 DNA Methylation

DNA methylation, the covalent addition of a methyl group to a cytosine residue, usually in a cytosine-guanine pair (CpG), is the most widely studied epigenetic

modification. CpGs are enriched in clusters referred to as CpG islands (CGIs), associated with the promoter regions of up to 60% of genes (Cedar and Bergman 2012). While various criteria can be used to define CGIs, originally they were described as regions larger than 200 base pairs with more than 50% GC content and an observed/expected ratio of CpGs greater than 0.6 (Wang and Leung 2004). While around 70% of CpGs are methylated in mammals, when clustered in CGIs they are generally unmethylated (Suzuki and Bird 2008), with DNA methylation in these islands typically associated with gene silencing. This transcriptional repression by DNA methylation is brought about by the recruitment of chromatin condensing proteins and to a lesser extent by physically blocking transcription factor binding (Cedar and Bergman 2012). Although CGIs have been well-established to co-localise with transcription start sites, a 2010 study found around half of CGIs are located away from transcription start sites, at non-traditional, poorly characterised promoters which they termed "orphan CGIs" (Illingworth *et al.* 2010; Jones 2012).

Less extensively studied but potentially more relevant to disease, is the regulatory role of methylation extending from CGIs and within gene bodies (intragenic), as these regions have been found to be more variably methylated between tissue types and in cancerous tissue compared to normal tissue (Irizarry *et al.* 2009; Gertz *et al.* 2011). Regions 2Kb either side of CGIs have been designated island shores and regions a further 2Kb outside as shelves (Irizarry *et al.* 2009). Methylation further away from these regions, in CpG poor areas has been termed "open seas" (Sandoval *et al.* 2011). For a schematic on the genomic landscape of DNA methylation see Figure 1.4.

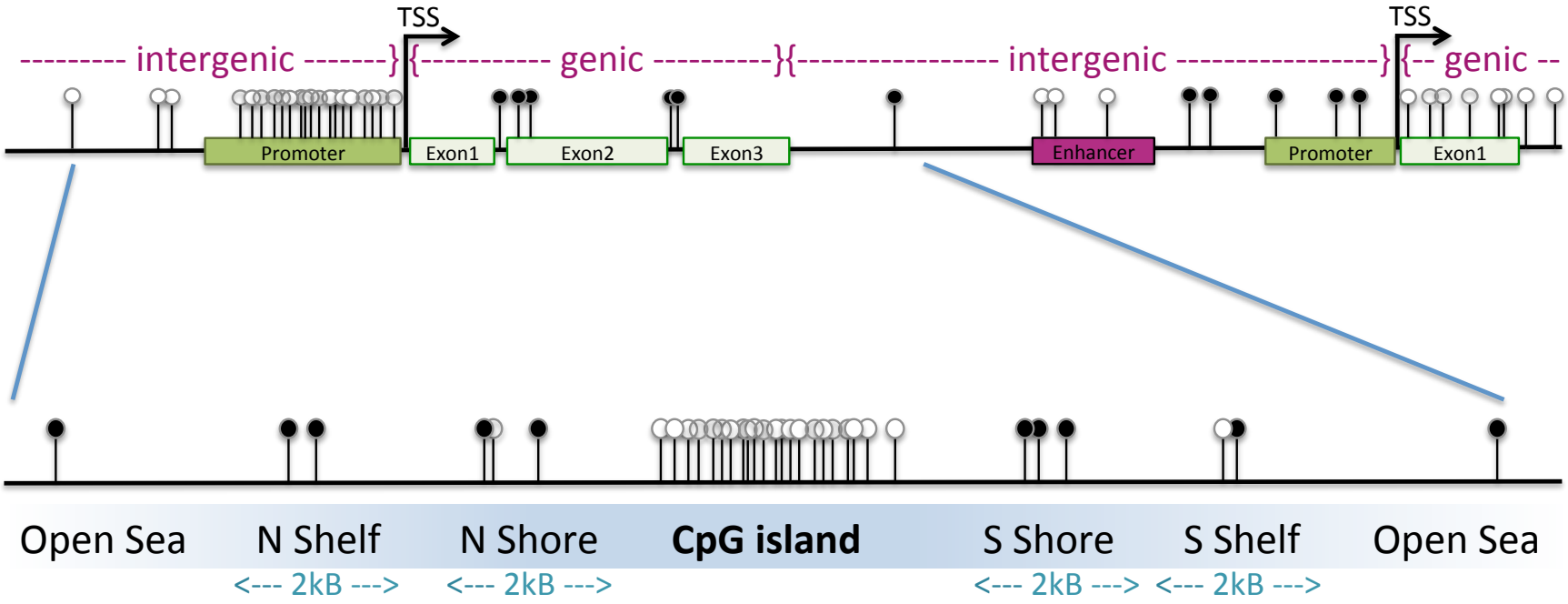


Figure 1.4 Depiction of the epigenetic landscape.
The upper panel provides a schematic of a typical intergenic and genic landscape including CpG distribution (lollipop figures) and methylation patterns (black for methylated, white for unmethylated) across exonic regions and promoter/enhancer regulatory regions. The transcription start site is indicated by the black arrows labelled 'TSS'. The lower panel depicts the most common annotation of genomic CpG density, with north (N) and south (S) shores 2kB either side of CpG dense islands and north (N) and south (S) shelves a further 2kB outside these. Further out, Open Seas denote CpG poor regions. Adapted from (Stirzaker *et al.* 2014).

Genome-wide, intragenic methylation patterns have been described as a bell-curve shape, with high methylation in genes of moderate expression levels and low methylation in genes with either high or low gene expression (Jjingo *et al.* 2012). These differences have been attributed to varying chromatin accessibility during transcription, with the idea being that at a low level of gene expression nucleosomes are tightly packaged and DNA methyltransferases (DNMTs) are unable to access CpG sites. At moderate levels of transcription, nucleosomes are repositioned to allow RNA polymerase II access to the DNA, coincidentally allowing DNMT access. However once RNA polymerase II density peaks with high gene expression, DNMT access is again restricted and methylation levels drop (Jjingo *et al.* 2012). Methylation at gene bodies may contribute to transcriptional elongation by inactivating alternate intragenic promoters or non-coding transcripts (Maunakea *et al.* 2010). This may help to explain the DNA methylation paradox, whereby methylation appears to have opposing effects on gene expression depending on whether it is present in promoter or gene body regions (Jones 1999; 2012).

1.5.4 Establishing and Maintaining Epigenetic Patterns

The establishment and maintenance of epigenetic patterns is influenced by a range of intrinsic and extrinsic factors including environment, diet, stochastic changes and the underlying genetic sequence (Aguilera *et al.* 2010; McKay *et al.* 2012) . The various epigenetic marks and mechanisms work collectively to create divergent epigenetic patterns across tissue and loci, which vary across populations and with age (ENCODE Project Consortium *et al.* 2012).

Environmental influences on epigenetic patterns include pesticides such as vinclozolin (Anway *et al.* 2005), which is used to kill fungal growth on crops but also acts as an endocrine disrupter in humans and is a known risk factor for prostate cancer, as discussed above. Another known environmental risk factor for prostate cancer, BPA a synthetic estrogen, has also been shown to influence DNA methylation patterns (Dolinoy, Huang and Jirtle 2007). There is evidence that at least some of these environmentally driven changes can be passed down through meiosis as transgenerational epigenetic inheritance (Guerrero-Bosagna and Skinner 2011; Daxinger and Whitelaw 2012; Veenendaal *et al.* 2013). However, the evidence for this inheritance and the mechanisms involved still remain to be fully elucidated, partly as they require escape of epigenetic reprogramming, a two phase process discussed below which would normally act to eliminate transmission of epigenetic marks down generations. Such inheritance is beyond the scope of this thesis; however a detailed discussion of the current understanding of transgenerational epigenetic inheritance is reviewed in Pembrey *et al.* (Pembrey *et al.* 2014).

While the effect of environment on epigenetic profiles, particularly DNA methylation is widely acknowledged, stochastic changes may in fact be more common than environmentally induced changes, with one study examining 4000 human genes in clonal cell lines, observing 300 to have random monoallelic expression (Gimelbrant *et al.* 2007). Epigenetic stochasticity can be defined as a combination of epigenetic variation in the germline and somatic instability. Similar to Richards' 'facilitated epigenetic variation' model (Richards 2006), Feinberg and Irizarry's 'inherited stochastic variation model' proposes genetic sequence variation underlies the

propensity for epigenetic variation, as certain DNA sequences are not only directly responsible for particular traits but also increase natural methylation variation for that trait (Feinberg and Irizarry 2010). Various stochastic and environmental factors then influence DNA methylation at these variably methylated regions (VMRs), resulting in increased phenotypic differences, which are then acted on by Darwinian selection in a similar manner to selection pressures affecting purely genetic traits. In subsequent studies they found the sites of greatest DNA methylation variability in colon cancer corresponded to the sites of greatest variability in other cancers including lung, breast and ovarian cancers, with these sites normally having distinct tissue specific DNA methylation patterns (Hansen *et al.* 2011). Thus heritable DNA methylation variation could provide some contribution to the unexplained heritable genetic component of common complex diseases.

Once DNA methylation patterns are established, DNA methyltransferases (DNMTs) ensure tissue specific methylation patterns are maintained through mitosis with high precision and fidelity (Cedar and Bergman 2012). In contrast, at a global level, methylation is substantially 'wiped clean' during gametogenesis to provide the developing embryo the capacity for totipotency and prevent accumulation of epigenetic changes from previous generations (Seisenberger *et al.* 2012). This epigenetic reprogramming occurs in two waves, firstly during pre-implantation and secondly after primordial germ cell migration, including the removal of imprinted marks (Sasaki and Matsui 2008).

Epigenetic reprogramming involves both active and passive demethylation, which

has long perplexed the scientific community due to their inability to identify the enzyme responsible for this demethylation. Insight into this process has been gained only recently following the discovery of the additional DNA modification, hydroxymethylation (5hmC), enriched in Purkinje cells in the brain (Kriaucionis and Heintz 2009) and also embryonic stem cells (Ito *et al.* 2010; Cimmino *et al.* 2011). The ten-eleven translocation (TET) family of proteins are responsible for this modification, and there is evidence that it is an intermediate in the process of active demethylation (Hackett *et al.* 2013). High levels of the modification, particularly in neural cells and during embryonic development suggests that it may have a yet to be elucidated role in regulating the genome. Interestingly, 5hmC levels have been shown to be reduced in cancers including prostate cancer (Haffner *et al.* 2011) and it has been suggested that this reduction may be an early event in prostate cancer development, possibly due to TET down-regulation as a result of environmental stress and aging (Chia *et al.* 2014).

1.5.4.1 Sequence driven methylation variation: meQTLs

Genomic sequence variation may have the greatest impact on epigenetic patterns, as one study examining a three generation family estimated that genotype explained around 80% of the variation in DNA methylation (Gertz *et al.* 2011). There has since been much research into 'methylation quantitative trait loci' or meQTLs; sequence variants across the genome that drive methylation patterns (Drong *et al.* 2013). These meQTLs have been mapped in a variety of tissues, stages of development, populations and across different organisms (Gibbs *et al.* 2010; Bell *et al.* 2011; Drong *et al.* 2013; Smith *et al.* 2014). MeQTLs can occur at the CpG site itself

(meSNP), in close proximity (*cis*) (Drong *et al.* 2013) or distantly (*trans*) (Lemire *et al.* 2015). Arguably the greatest effect on methylation is mediated through meSNPs as these can directly create or remove a CpG site, immediately influencing methylation at that site and possibly in neighbouring regions. Indeed Zhi and colleagues found that two thirds of the strongest meQTL signals in their study were due to meSNPs and that 80% of generic variants at meSNPs were meQTLs (Zhi *et al.* 2013). They also found that the CpG disrupting SNP significantly affected methylation at CpG sites within 45bp and continued up to 10kb, likely in associated with linkage disequilibrium. Providing further evidence for the importance of meSNPs, a recent study examining publicly available data found 23% of common variants were meSNPs and that these SNPs were significantly enriched in meQTLs and more likely to be trait-associated cancer loci (Zhou *et al.* 2015).

DNA sequence variants not only drive methylation changes but also histone modifications, with three recent studies pointing to an important role of genetic variants in determining histone modification patterns (McVicker *et al.* 2013; Kilpinen *et al.* 2013; Kasowski *et al.* 2013). In these studies, hundreds of variants were associated with changes to histone modifications and gene expression, with the underlying mechanism thought to be altered transcription factor binding. Various other mechanisms have been proposed for the altered gene expression associated with meQTLs, including altered binding of proteins such as the CTCF transcription factor which has different affinities for methylated and unmethylated DNA (Shukla *et al.* 2011), and altered transcription elongation, splicing and recombination rates (Olsson *et al.* 2014).

1.5.5 Genetically driven Epigenetic Disease Susceptibility

Inherited genetic variants have been shown to drive methylation changes in disease with meQTLs associated with osteoarthritis (Rushton *et al.* 2015), neuropsychiatric diseases (Taqi *et al.* 2011) and complex diseases such as diabetes and cancer. A recent study examining 40 previously identified diabetes risk SNPs found nearly half introduced or removed a CpG site (Dayeh *et al.* 2013), while an extensive genome-wide association study between genetic variation, methylation patterns, mRNA expression and insulin secretion identified SNP-CpG pairs in several genes involved in proliferation and apoptosis of pancreatic ß-cells (Olsson *et al.* 2014).

DNA methylation was the first epigenetic mark to be linked to cancer (Feinberg and Tycko 2004) in the early 1980s and since then the global hypomethylation and regional specific hypermethylation has been well characterised in tumour tissue. Global hypomethylation due to loss of methylation at repetitive sequences and retrotransposons often occurs in prostate cancer (Yegnasubramanian *et al.* 2008) with the level of hypomethylation correlated to cancer severity (Bedford and van Helden 1987). In contrast, hypermethylation has been particularly seen at promoter CGIs of tumour suppressor genes such as *Glutathione S-Transferase Pi 1 (GSTP1)*, *Ras Association Domain Family Member 1 (RASSF1A)* and *O^6-alkylguanine DNA alkyltransferase (MGMT)* where aberrant methylation leads to gene silencing (Song *et al.* 2002; Kang *et al.* 2004). More recent cancer research efforts have focussed on methylation at shores and shelves where most (up to 76%) tissue-specific differential methylation occurs and interestingly, the majority of differential methylation changes in cancer also occur at these regions (Irizarry *et al.* 2009).

A key example of a genetically driven methylation abnormality in the cancer field is Lynch syndrome; an autosomal dominant cancer susceptibility condition where two thirds of cases result from heterozygous loss-of-function mutations in DNA mismatch repair genes, most commonly *MutL Homolog 1 (MLH1)* and MutS Homolog 2 *(MSH2)* (Ward *et al.* 2013). However, such mutations are not apparent in around one-third of Lynch Syndrome cases, some of which (~4% for *MLH1* (Ward *et al.* 2013)) can be explained by epimutations in *MLH1* and *MSH2*. These epimutations lead to transcriptional inactivation of the gene, essentially having the same effect as a genomic sequence mutation seen in other Lynch syndrome cases. One possible mechanism underlying these epimutations involves 'primary' DNA methylation changes independent of any sequence change, resulting in labile epimutations, which can be reversed in the germline and are therefore inherited in an unpredictable, non-Mendelian manner or not passed on at all.

Alternatively, 'secondary' epimutations may result from underlying sequence changes, including promoter deletions and SNPs (Hitchins and Lynch 2014). For example, the c.-27C>A germline variant in the 5'UTR of the *MLH1* gene has been linked to cancer susceptibility through transcriptional silencing (Hitchins *et al.* 2011). In these cases the disease follows a more predictable inheritance pattern as the epimutation is driven by a genetic variant. As yet undiscovered sequence mutations may also be the underlying carcinogenic mechanism in subsets of cancers such as Cowden syndrome, where some individuals have hypermethylation epimutations in the absence of known sequence mutations (Bennett, Mester and Eng 2010).

Underlying genetic drivers have also been linked to epimutations in sporadic cases of renal cell cancer (RCC) where SNPs were associated with promoter hypermethylation of the *Von Hippel-Lindau Tumor Suppressor* (*VHL)* gene in tumour tissue, a gene previously shown to be genetically altered in individuals with the familial form of the cancer (Moore *et al.* 2011). Similarly, in colorectal cancer, a C>T point mutation at an enhancer element of the mismatch repair gene *MGMT,* has been linked to aberrant promoter methylation and gene silencing (Ogino *et al.* 2007). Finally, Shen *et al* (Shen *et al.* 2013) have demonstrated that susceptibility SNPs at the *HNF1 Homeobox B (HNF1B)* locus in ovarian cancer are associated with altered methylation and consequent expression of *HNF1B.*

Evidence for genetically driven aberrant methylation patterns in prostate cancer is provided by a 2013 study examining DNA methylation changes in prostate cancer metastases (Aryee *et al.* 2013). While the authors found substantial methylation alteration differences between individuals, methylation changes were often maintained in the same individual across various anatomically distinct metastases compared to matched normal non-prostatic tissue, in a similar manner to copy number alterations. These abnormally methylated regions within the same individual across metastases were often located in cancer-related genes, as were regions that were similarly hypermethylated between individuals. Despite less intra-individual tumour methylation heterogeneity than between individuals, clonal evolution still created noticeable intra-individual methylation differences between metastases. The authors attributed this mainly to stochastic factors as these tumours pass through a "very narrow individual-specific clonal gate" after which minimal heterogeneity

occurs. In a more loci-specific study, Ianni *et al.* examined genetically driven methylation alterations in several genes including *Glycine N-Methyltransferase (GNMT),* a gene coding for an enzyme involved in regulating synthesis and availability of methyl groups. The T allele genotype at one SNP in the *GNMT* promoter was linked to altered DNA methylation, decreased gene expression and heightened prostate cancer risk (Ianni *et al.* 2012). Together these studies point to the importance of examining genetically driven methylation changes in further understanding prostate cancer development.

Thus there is now considerable interest in mapping inherited methylation changes influencing disease susceptibility and disease course. Given the recent technological advances that are enabling integration of genetic, epigenetic and phenotypic data it is likely that more examples of diseases resulting from genetic drivers of epigenetic change will be described in the future.

1.5.6 The promise of epigenetic diagnosis and therapy

Aberrant methylation has been suggested as a diagnostic and prognostic marker for various cancers for nearly a decade. *GSTP1*, the most commonly altered gene in prostate cancer, improves specificity and sensitivity of PSA diagnosis, particularly on recurrence (Hopkins, Burns and Routledge 2007; Woodson *et al.* 2008; Maldonado *et al.* 2014), while promoter methylation at the *MGMT* gene predicts response to chemotherapy in glioblastoma (Wiewrodt *et al.* 2007). More recently, a 2012 prospective study of cervical neoplasia further demonstrated the utility of harnessing our heightened understanding of epigenetic predisposition in improving

diagnostic tests. Examining DNA methylation variability between epithelial samples predicted the risk of cancer three years prior to morphological changes (Teschendorff and Widschwendter 2012; Teschendorff *et al.* 2012). Feinberg suggests that using such tests to identify subgroups for further traditional follow-up screening could increase the positive predictive value of these more invasive and expensive tests by over 90% and greatly reduce cancer deaths (Feinberg 2014).

Furthermore, epigenetic therapy offers the exciting possibility of resetting gene expression in a way not yet possible for genomic mutations, since unlike genetic variation and chromosomal defects that permanently alter the genome, epigenetic patterns are more dynamic and pharmaceutically adaptable. Several epigenetic drugs have been approved for treatment of other cancers in the USA, with many more in clinical trials (Bojang and Ramos 2014). However, these drugs are designed to correct the epigenetic alterations acquired during disease development (Sharma, Kelly and Jones 2010) with the assumption that these acquired epigenetic alterations are driven by the disease process itself. More recently it has been hypothesised that inherited genetic variation can drive epigenetic alterations and further, that these contribute to disease susceptibility or disease course. A greater understanding of these drivers of disease would allow for more effective, personalised medical treatment of prostate cancer.

1.6 Project Rational

The focus of this study is to adopt a new approach to elucidate the underlying molecular mechanisms driving prostate cancer development by using the Tasmanian

Familial Prostate Cancer resource to examine inherited determinants of methylation patterns. This resource contains genealogical and pathology records for over one thousand people from fifty large families, with archived blood, buccal and prostate tissue samples for individuals diagnosed with prostate cancer and their close relatives.

1.6.1 Hypothesis

The hypothesis for this study is that DNA sequence changes in non-coding regions of the genome can alter transcriptional activity, triggering epigenetic changes in regulatory regions, which lead to gene silencing and contribute to prostate cancer development and progression. It is hypothesised that these can be identified by examining germline DNA of individuals from families exhibiting a dense aggregation of prostate cancer cases.

1.6.2 Aims

Two broad aims were developed to examine this hypothesis:

Aim 1: To utilise the Tasmanian Familial Prostate Cancer Resource, containing dense aggregations of prostate cancer cases, to examine the association between genotype, as measured by SNPs, and epigenotype, as measured by DNA methylation.

Aim 2: To utilise the Tasmanian Familial Prostate Cancer Resource and the Tasmanian Prostate Cancer Case Control Study to examine the association between epigenotype and prostate cancer occurrence.

Chapter 2 – Features of the Tasmanian Familial Prostate Cancer Resource guiding study design

2.1 Introduction

2.1.1 Employing familial data to examine disease-relevant meQTLs

Population-based GWAS studies have successfully identified part of the common variation linked to disease, yet the effect size of variants is often small and difficult to verify (Hindorff *et al.* 2009; Marjoram, Zubair and Nuzhdin 2014). As it becomes clearer that common SNPs do not explain a large portion of the genetic risk of complex disease, rare variants are being examined in more detail to help further understand inherited risk of disease (Manolio *et al.* 2009; Wijsman 2012). Family studies provide one of the best opportunities to examine rare deleterious variants, as these variants are often enriched in families with clusters of affected individuals (Curran, Meikle and Blangero 2011). The inclusion of pedigree structures in these studies also provides additional analysis advantages, including imputation for family members where genetic material is unavailable and interpretation and ranking of disease segregating variants (Thomas 2012). Decreased genetic complexity combined with shared environments also allow for greater confidence in detecting true genetic differences over confounding factors (Pattaro and Saint-Pierre 2013). Additionally, certain statistical challenges are reduced in familial studies; pedigrees inherently lack population stratification which can create off target associations, and a much smaller sample size is required to achieve the same power, assuming the

pedigree has an appropriate trait distribution. There is a lower risk of false positives as less statistical tests are required once specific regions of interest are identified (Wijsman 2012).

The power of familial study designs can be observed in the recent success of two familial studies identifying functional variants linked to disease, one examining autism risk and the other renal function (Marchani *et al.* 2012; Park *et al.* 2013). This utility is further exemplified by the divergent results of two late-onset Alzheimer's disease studies, one familial and the other case-control. While both studies evaluated the same four candidate genes, the success of each study was markedly different. The population-based case-control study of 17,313 individuals, had a sample size forty times greater than the familial approach, yet was unable to find any significant disease-linked SNPs (Gerrish *et al.* 2012). Contrastingly, the familial study which sequenced pooled samples from 449 unrelated affected individuals from families with clusters of Alzheimer's disease, identified numerous disease-segregating rare variants (Cruchaga *et al.* 2012).

Similar success has been seen in familial studies examining prostate cancer risk, for example in the case of *Homeobox B13* (*HOXB13)* as discussed in Chapter 1 under section 1.3.1. In this case, family studies were key to discovering a risk region on chromosome 17 and also subsequent elucidation of the disease-segregating mutation. Unlike the above Alzheimer's disease case, possibly due to the heterogeneous nature of prostate cancer predisposition, *HOXB13* was later validated in population based studies and found to explain around 1% of the familial risk of

prostate cancer in the UK population (Kote-Jarai *et al.* 2015). Additionally, work from our laboratory using the same familial prostate cancer data set as utilised in the current project, discovered novel risk SNPs associated with *Integrin alpha 2* (*ITGA2)*, a gene not previously associated with prostate cancer risk (FitzGerald *et al.* 2009).

While familial studies have traditionally been applied to analyse genetic risk for complex diseases, advances in technology and analysis tools have recently allowed for epigenetic risk to also be investigated. In a similar manner to examining the effect of genetic variation on gene expression or quantitative traits (eQTLs/QTLs), the effect of genetic variation on methylation and subsequently gene expression and quantitative traits (meQTLs) can now also be explored, using similar laboratory techniques and data analysis algorithms (Gaunt *et al.* 2016). Sequence and array-based technologies are now capable of measuring epigenetic variation as well as genetic variation, with DNA methylation a particular disease focus.

The majority of these next generation methylation techniques depend upon a key innovation from the Kanematsu laboratories in 1992, bisulphite conversion (Frommer *et al.* 1992). The process takes advantage of the selective deamination by bisulphite of cytosine residues over other nucleotides. Critically, methylation of the cytosine residue protects it from this conversion and as such methylated residues can be distinguished from those that are unmethylated. Unmethylated cytosines are converted to uracil and then thymine during PCR amplification, as demonstrated in Figure 2.1 (Patterson *et al.* 2011).

Recent studies have demonstrated the utility of exploiting familial resources to examine the effect of meQTLs in both *cis* and *trans* (Lemire *et al.* 2015; Kulkarni *et al.* 2015). The current study similarly aims to harness the power of a familial study design in combination with next generation epigenetic technologies to examine the association between genotype, epigenotype and prostate cancer occurrence. Whilst this strategy has now been adopted by several researchers in the field of familial genetics, at the commencement of this project, this approach was new to the field. To achieve this larger aim, two specific objectives first needed to be addressed. Firstly, clusters of individuals from the Tasmanian Familial Prostate Cancer Resource with high disease burden needed to be identified, and secondly high quality genome-wide genotype and methylation data needed to be generated from these individuals.

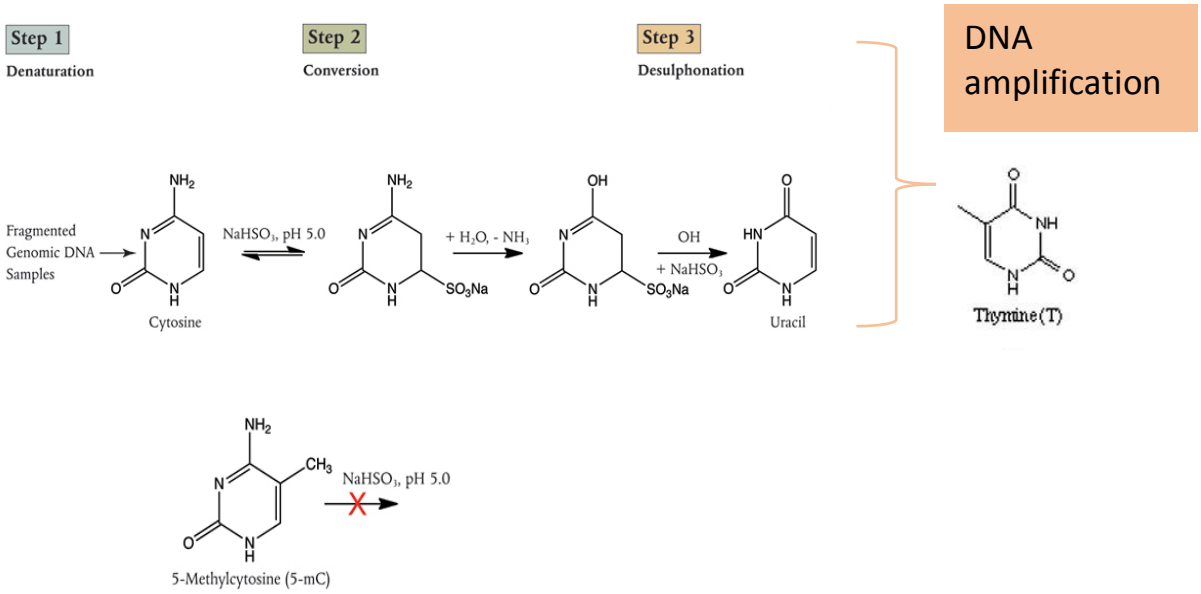


Figure 2.1 Bisulphite Conversion of DNA
Genomic DNA is denatured (Step 1) before unmethylated cytosines undergo bisulphite conversion with sodium bisulphite (Step 2). A desulphonation step removes the sulphite moiety, generating uracil (Step 3). During subsequent PCR amplification uracil is replaced with thymine. Alternatively, the methyl group protects methylated cytosines from bisulphite conversion and these nucleotides remain as cytosines, and can thus be distinguished from unmethylated cytosines which have been converted to thymine. Adapted from New England BioLabs EpiMark® Bisulfite Conversion Kit Protocol.

2.2 Sample Selection Strategy and Evaluation of Data Quality

2.2.1 Sample Selection from the Tasmanian Familial Prostate Cancer Resource

The Tasmanian Familial Prostate Cancer Resource provides a rare opportunity to further examine the underlying mechanisms of prostate cancer development. It was initially established in the 1990s from large families with multiple cases of prostate cancer. The families were identified from the records of the Tasmanian Cancer Registry. As PSA testing had not been widely adopted in Tasmania at the time, these original affected men were recruited to the study having presented with a symptomatic form of the disease, with all cases being histologically confirmed. The Tasmanian Cancer Registry and the Menzies Institute for Medical Research genealogical database has since been utilised to expand this original database to include additional men diagnosed with prostate cancer as well as their close relatives. Many families extend to five or six generations. This unique resource now contains genealogical and pathology records for over one thousand people from sixty large families. As well as whole blood and buccal samples, over 75% of cases have archived pathology specimens in the form of paraffin embedded prostate tissue. These are stored at one of three locations; pathology laboratories at the Royal Hobart Hospital, the Cancer Genetics laboratory at the Menzies Institute for Medical Research and private pathology laboratories in Hobart.

The current study focuses on four large families from this resource, Prostate Cancer Families 9, 11, 22 and 72, with abridged pedigrees shown in Figures 2.2-5. For this study sixty-two individuals were selected primarily from these four large families and their DNA subjected to genome-wide methylation and / or genotype analysis.

Selection of individuals was based on disease burden, as determined by the number of affected first and second degree relatives and the proband's age at diagnosis. Selected individuals were of Caucasian descent, ranging in age from 23 to 89 years and represented clusters of densely aggregated cases of affected men and close relatives. These individuals included thirty-four men affected by prostate cancer, fifteen unaffected men, ten females (including one diagnosed with another cancer) and three men diagnosed with another form of cancer. Female samples were chosen if they had at least one first degree relative affected by prostate cancer. These samples were used to provide additional statistical power in the genotype-methylation analysis described in Chapter 4.

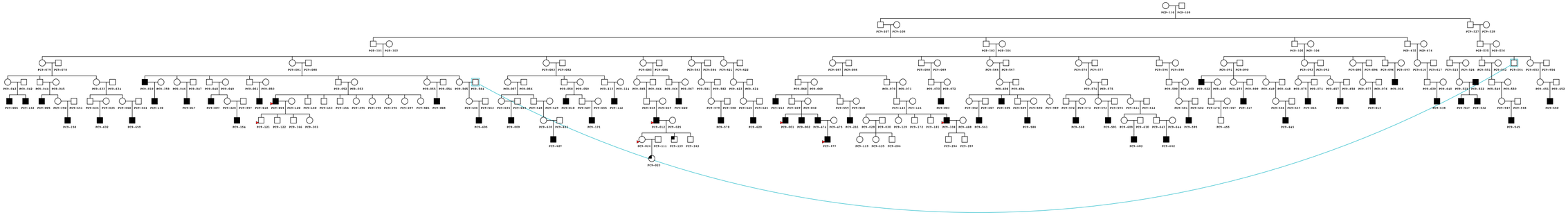


Figure 2.2 Tas PCa Family 9 abridged pedigree.
A section of the larger pedigree for Family 9 highlights the high rate of prostate cancer within the family, particularly in the later generations. Males are shown as squares and females circles. Men affected by prostate cancer are represented by filled in black squares, unaffected men by unfilled squares and individuals diagnosed with other cancers are indicated by a small filled box in the top left corner. A subset of samples analysed for methylation data are indicated by red arrow heads. Family 9 extends over nine generations and contains fifty-three men affected by prostate cancer. The blue line indicates where the same individual has been included twice in the pedigree, once in his own lineage on the far right and again towards the left of the pedigree where he has married a woman who has descended from the left branch of the family.

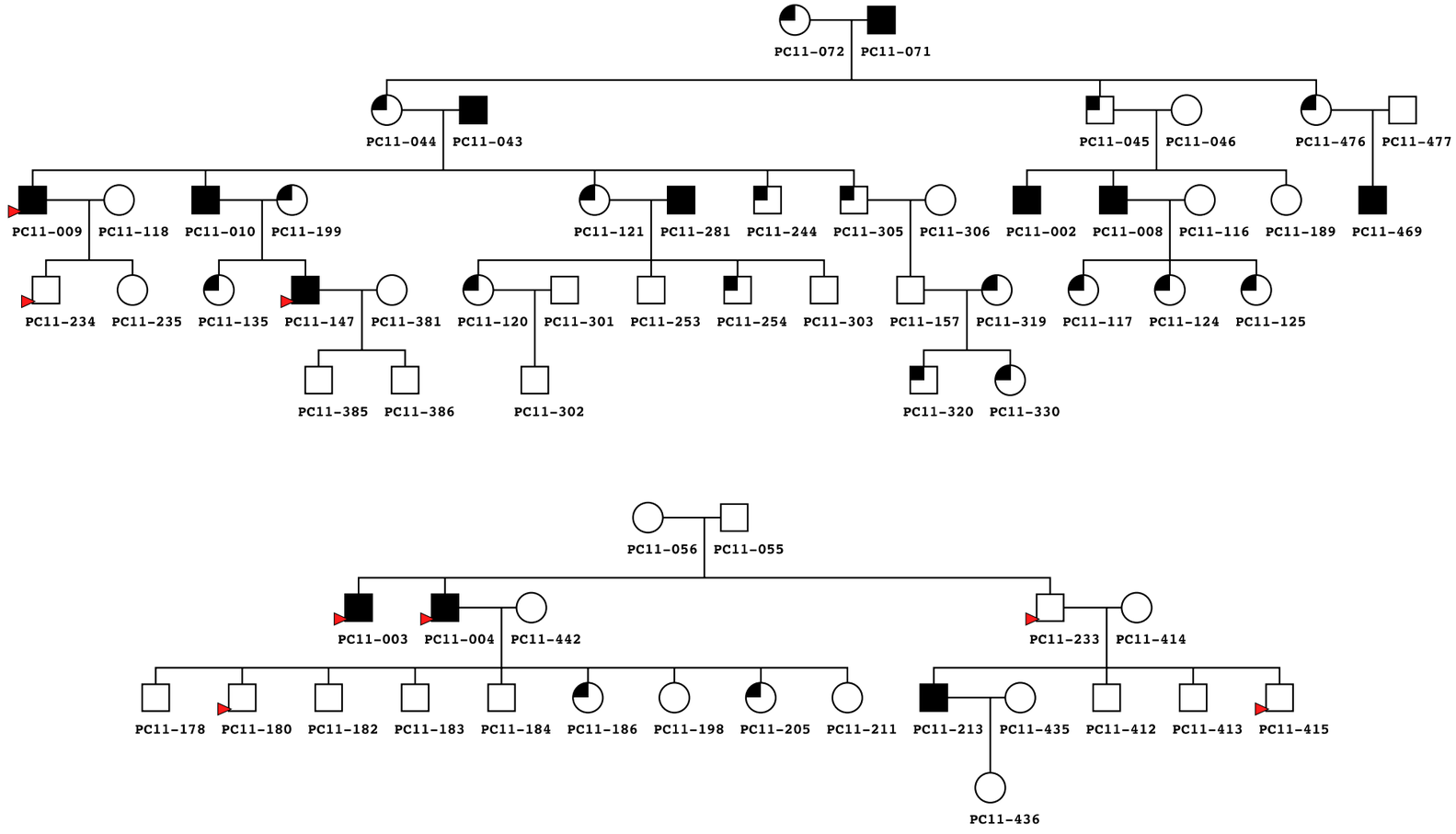


Figure 2.3 Tas PCa Family 11 abridged pedigree.
Two branches of the larger pedigree for Family 11 are illustrated, with men represented as squares and females circles. Men affected by prostate cancer are filled in black, unaffected men are unfilled and individuals diagnosed with other cancers are indicated by a small filled box in the top left corner. A subset of samples analysed for methylation data are indicated by red arrow heads. These abridged pedigrees for Family 11 cover five generations and include twelve affected men.

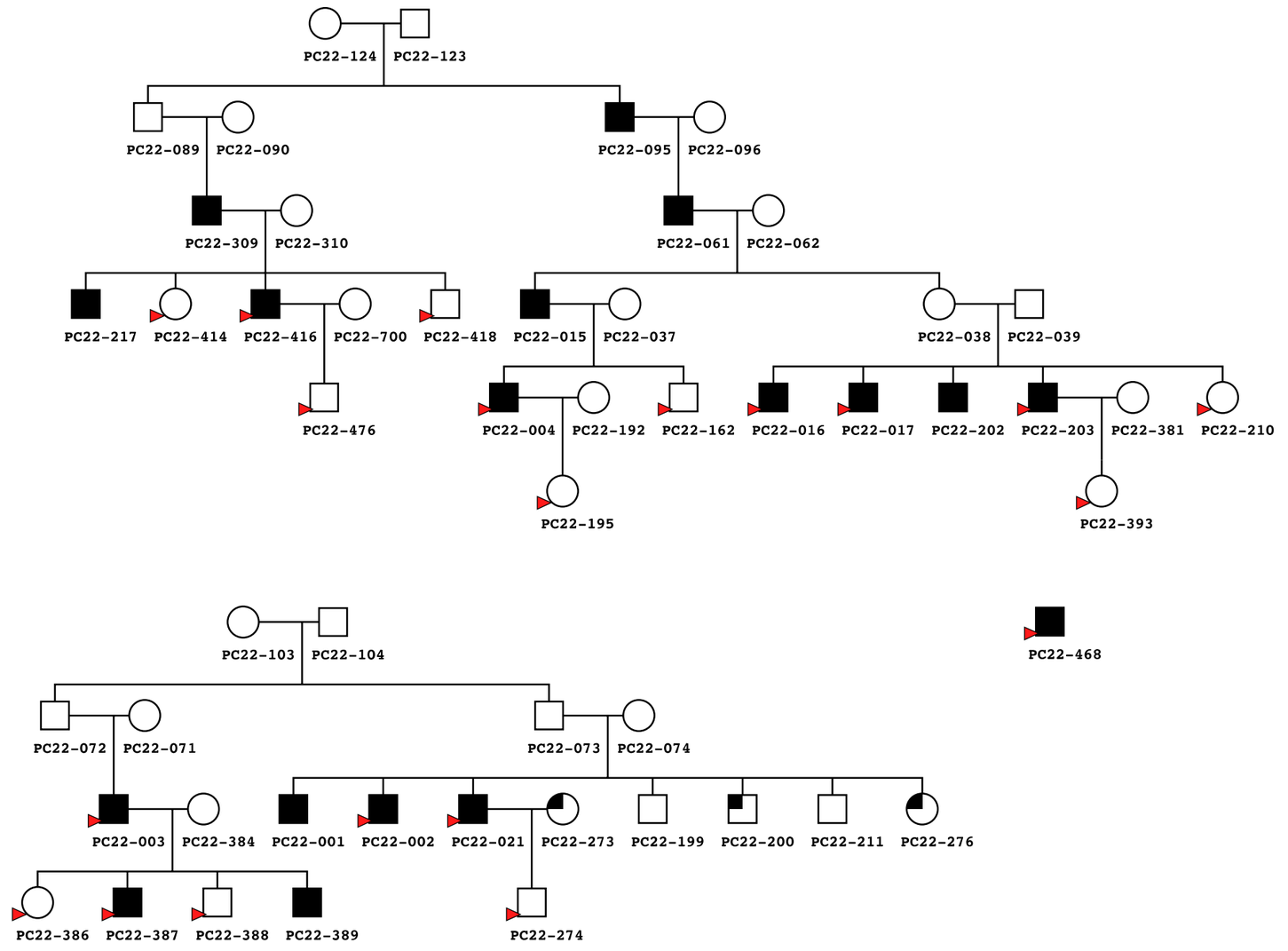


Figure 2.4 Tas PCa Family 22 abridged pedigree.
Two branches of the larger pedigree for Family 22 are illustrated, with men represented as squares and females circles. Men affected by prostate cancer are filled in black, unaffected are unfilled and individuals diagnosed with other cancers are indicated by a small filled box in the top left corner. A subset of samples analysed for methylation data are indicated by red arrow heads. The abridged Family 22 pedigrees cover six generations, including eighteen affected men.

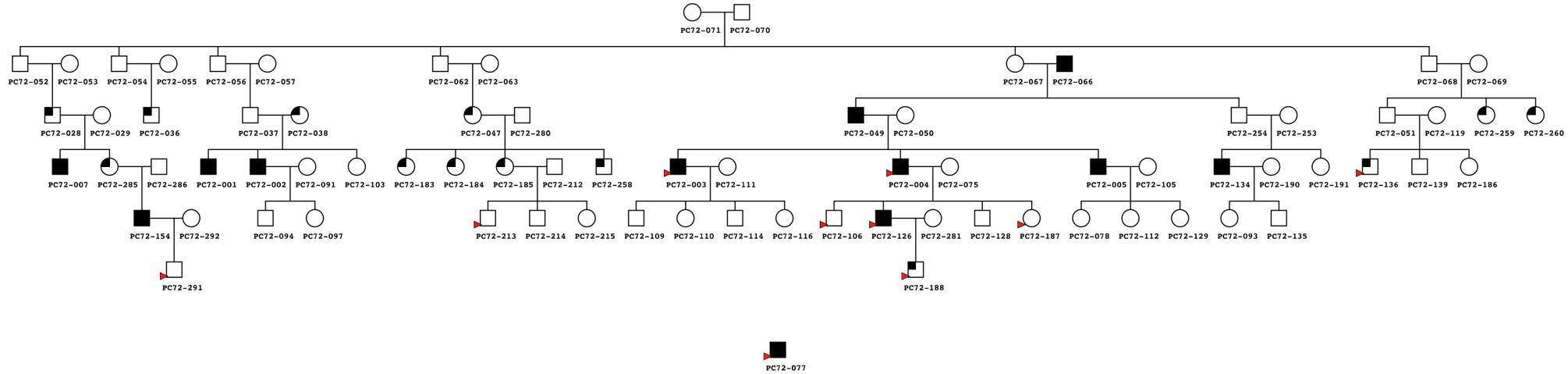


Figure 2.5 Tas PCa Family 72 abridged pedigree.
A section of the larger pedigree for Family 72 is illustrated, with men represented as squares and females circles. Men affected by prostate cancer are filled in black, unaffected are unfilled and individuals diagnosed with other cancers are indicated by a small filled box in the top left corner. A subset of samples analysed for methylation data are indicated by red arrow heads. The abridged family 72 pedigree depicts twelve affected men over six generations.

Overall, twenty-seven individuals selected had at least two first-degree relatives affected by prostate cancer, and often had three or four affected relatives (ten samples). Twenty-six had one affected first-degree relative and only eight had none, of which six had prostate cancer themselves. In addition to the number of first degree relatives affected by prostate cancer, samples were also selected to generate clusters of closely related individuals, with fourteen father-son pairs, seven brother pairs and six sibling trios included. One three-generation cluster from Family 72 was also selected, comprising two first generation affected brothers, three second-generation children (two men affected by prostate cancer and their sister) and a third generation male affected by another cancer. Figure 2.6 A-D displays the familial clusters (from families 9, 11, 22 and 72) and abridged pedigrees for the individuals selected for analysis. Some additional individuals were chosen for analysis from smaller pedigrees to ensure statistical power was maintained. These individuals were also prioritised on the number of first degree relatives affected by prostate cancer, and are shown in Figure 2.6 E.

A)

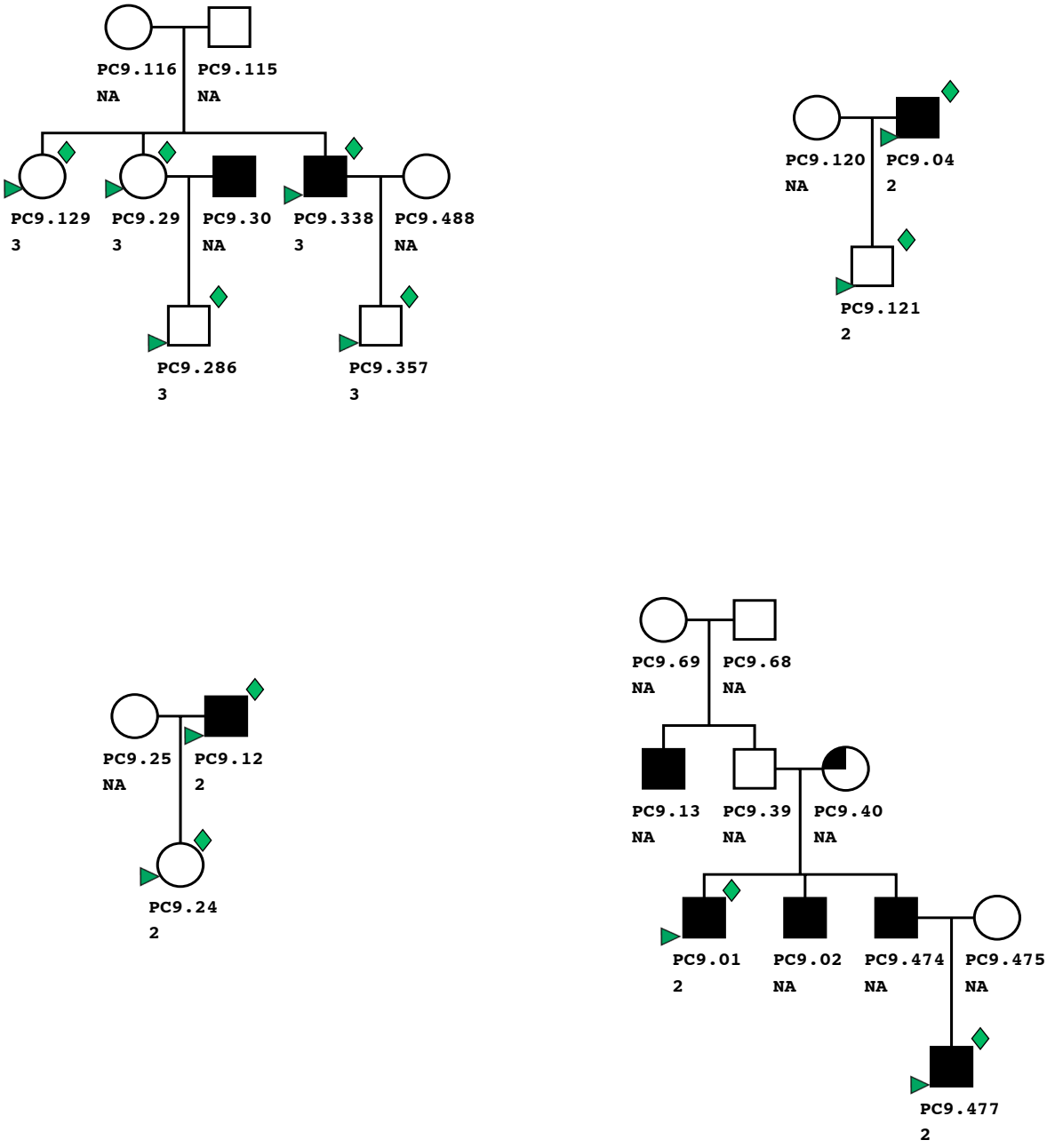


Figure 2.6 Pedigree clusters from the Tasmanian Familial Prostate Cancer study.
A) Four clusters from Family 9 were selected for analysis. Circles represent women and squares men, with individuals affected by prostate cancer filled in black, those unaffected unfilled and individuals affected by other cancers quarter filled. Samples with methylation data are indicated by an arrow-head, green for good quality samples and red for poor quality. Samples with genotype data are indicated by diamonds, green for good quality and red for poor quality. Replicate samples are indicated by square brackets around the sample name, and the batch for methylation array data is indicated underneath the sample name.

B)

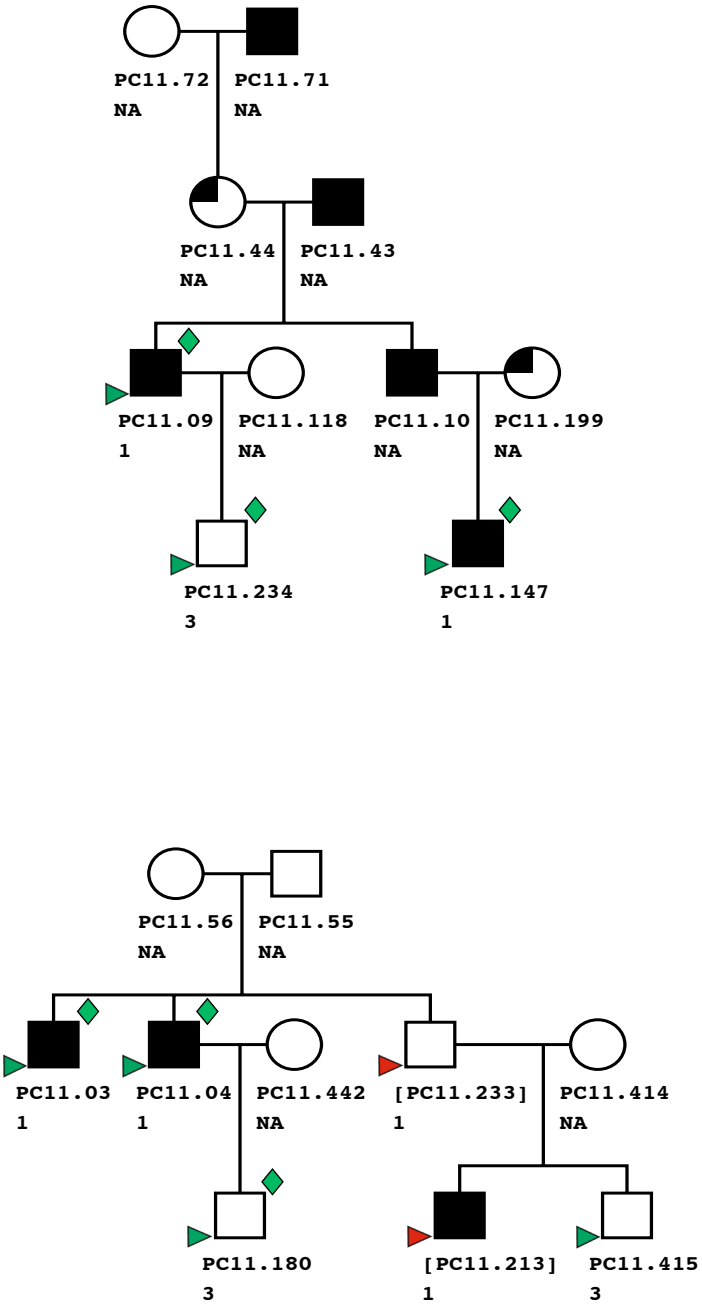


Figure 2.6 Pedigree clusters from the Tasmanian Familial Prostate Cancer study. B) Two clusters from Family 11 were selected for analysis. Circles represent women and squares men, with individuals affected by prostate cancer filled in black, those unaffected unfilled and individuals affected by other cancers quarter filled. Samples with methylation data are indicated by an arrow-head, green for good quality samples and red for poor quality. Samples with genotype data are indicated by diamonds, green for good quality and red for poor quality. Replicate samples are indicated by square brackets around the sample name, and the batch for methylation array data is indicated underneath the sample name.

C)

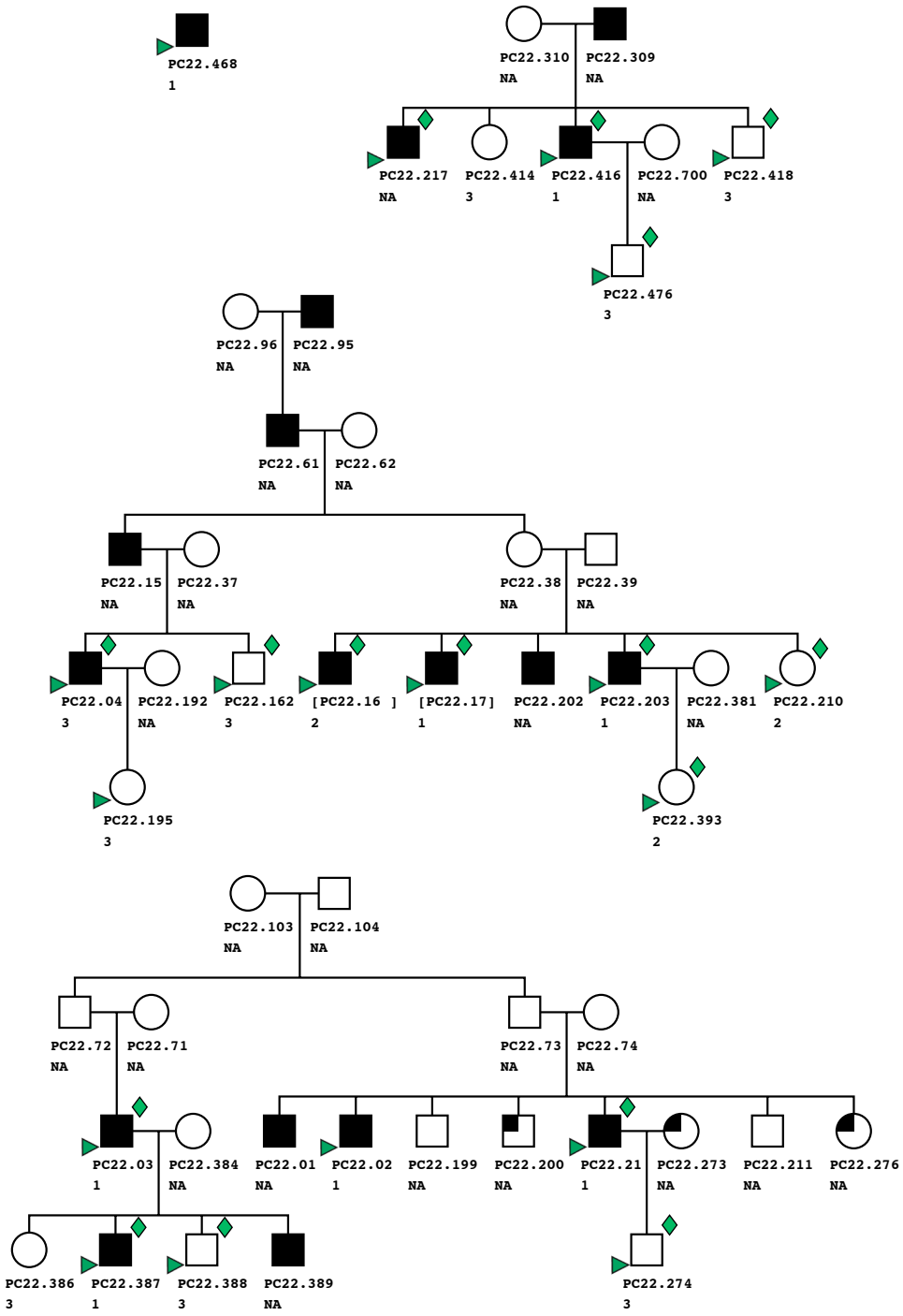


Figure 2.6 Pedigree clusters from the Tasmanian Familial Prostate Cancer study.
C) Three clusters from Family 22 were selected for analysis. Circles represent women and squares men, with individuals affected by prostate cancer filled in black, those unaffected unfilled and individuals affected by other cancers quarter filled. Samples with methylation data are indicated by an arrow-head, green for good quality samples and red for poor quality. Samples with genotype data are indicated by diamonds, green for good quality and red for poor quality. Replicate samples are indicated by square brackets around the sample name, and the batch for methylation array data is indicated underneath the sample name.

D)

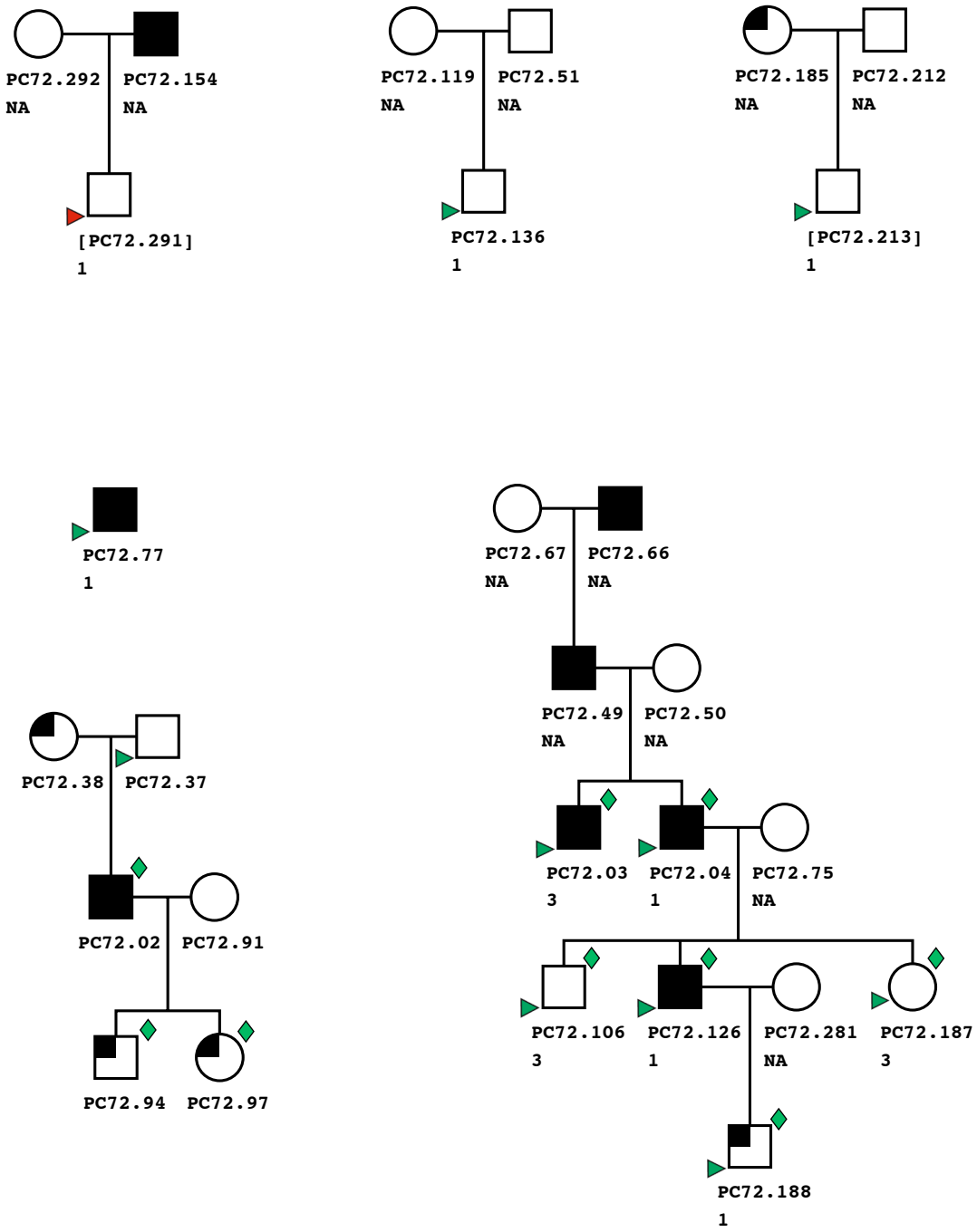


Figure 2.6 Pedigree clusters from the Tasmanian Familial Prostate Cancer study. D) Five clusters from Family 72 were selected for analysis. Circles represent women and squares men, with individuals affected by prostate cancer filled in black, those unaffected unfilled and individuals affected by other cancers quarter filled. Samples with methylation data are indicated by an arrow-head, green for good quality samples and red for poor quality. Samples with genotype data are indicated by diamonds, green for good quality and red for poor quality. Replicate samples are indicated by square brackets around the sample name, and the batch for methylation array data is indicated underneath the sample name.

E)

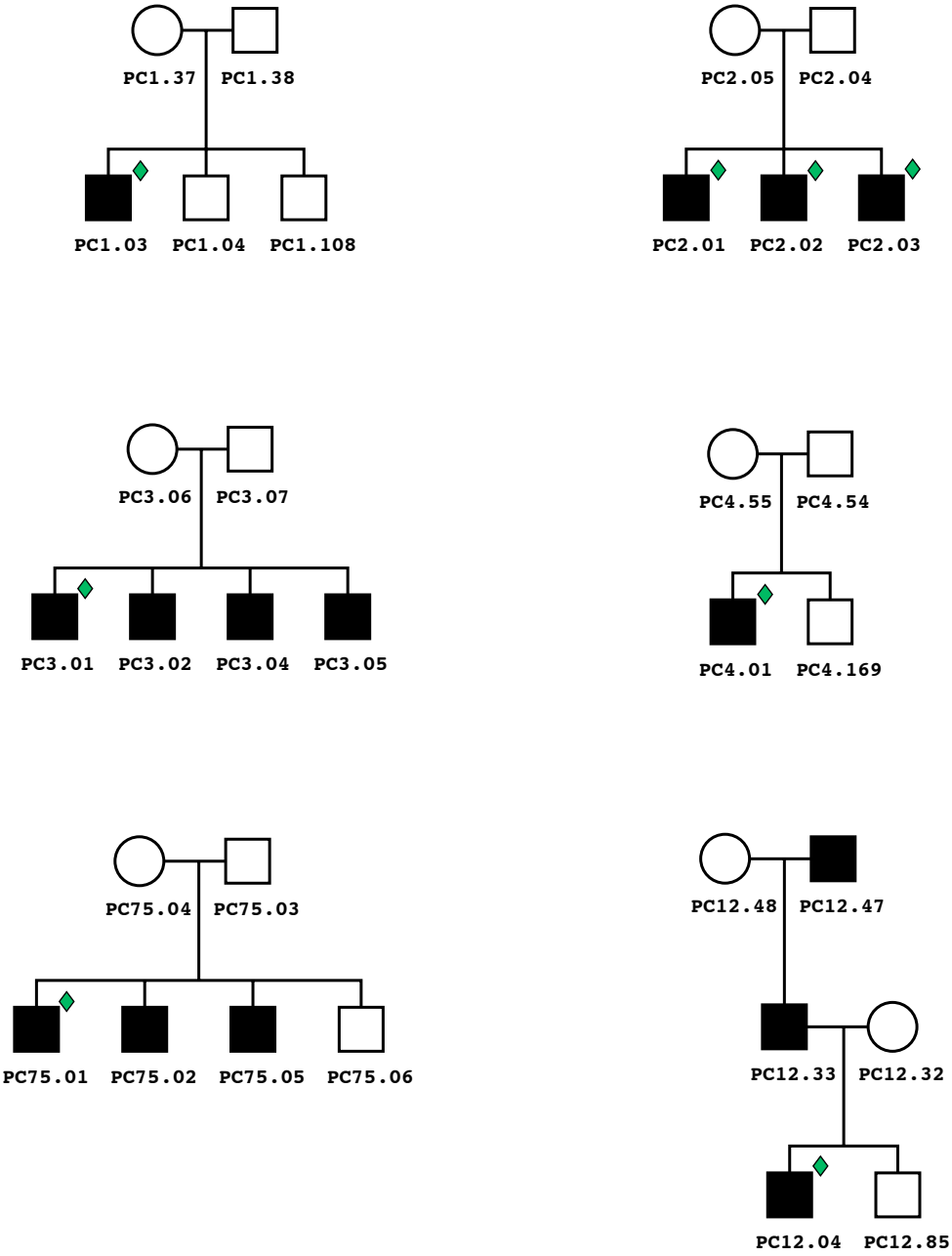


Figure 2.6 Pedigree clusters from the Tasmanian Familial Prostate Cancer study.
E) One cluster from each of families 1, 2, 3, 4, 12 and 75 were selected for analysis. Circles represent women and squares men, with individuals affected by prostate cancer filled in black, those unaffected unfilled and individuals affected by other cancers quarter filled. Samples with methylation data are indicated by an arrow-head, green for good quality samples and red for poor quality. Samples with genotype data are indicated by diamonds, green for good quality and red for poor quality. Replicate samples are indicated by square brackets around the sample name, and the batch for methylation array data is indicated underneath the sample name.

2.2.2 DNA isolation, preparation and initial quality control

Peripheral blood samples were collected with informed consent obtained from all participants, following ethics approval from the University of Tasmania (H9999, Human Research Ethics Committee Tasmanian Network). DNA was extracted from whole blood using the Nucleon BACC3 (GE Healthcare) DNA extraction kit, following the manufacturer's instructions. DNA was initially quantified on the Nanodrop 8000 (Thermo Scientific) and samples with a 260:280 ratio of less than 1.80 were further purified using the Zymo Clean & Concentrator (TM)-5 Kit (Zymo). DNA was then quantified using a Qubit® Flourometer. To ensure selected samples had sufficient quantity and quality of DNA to perform methylation and genotype arrays, each sample was quantified on the Nanodrop 8000 (Thermo Scientific) and samples with sufficient DNA quantity (at least 1.5μg) and purity ($A_{260/280}$ above 1.8) were interrogated via electrophoresis on a 2% agarose gel to further assess DNA quality. Samples with poor quality DNA, as indicated by small DNA fragments detected by gel electrophoresis, were excluded from further analysis.

2.2.3 Genome-wide DNA methylation analysis

The launch of Illumina's Infinium HumanMethylation 450k BeadChip (hereafter referred to as the 'methylation array') in 2012 has been central to the rapid advance in our understanding of epigenetic risk factors of disease. This cost-effective platform delivers a single "snapshot" of over 480,000 CpG sites throughout the genome. Adopted by many researchers, it ensured the continued popularity of epigenomic array-based technology, as earlier array technology lacked the same depth of genomic coverage (ie. the previous Illumina methylation array contained

only 27,000 CpG sites) and bisulphite sequencing costs at the time were prohibitive and indeed still surpass the cost and ease of laboratory processing of the methylation array.

Briefly, the methylation array involves denaturing and bisulphite converting DNA, followed by amplification and hybridization to a bead chip containing hapten labelled dideoxynucleotides. There are two types of probes on the array; one quarter are Infinium I probes which use a single colour channel but have a different bead for methylated or unmethylated DNA, while the remaining 75% are Infinium II probes, a novel design from the previous 27k array, with a single bead, using two different colour signals for detection, green for methylated (labelled with biotin) and red for unmethylated (labelled with 2,4-dinitrophenol). While Infinium II probes require only one bead per locus, increasing genomic coverage, they can only tolerate a maximum of three CpGs in the probe body. Infinium I probes can tolerate more CpGs, allowing for coverage of CpG dense regions.

While this technology is powerful and reproducible there are several factors that must be considered in experimental design and quality control to ensure reliable, high quality data is generated and ensure observed differences are true biological differences, not simply technical bias (Sun *et al.* 2011; Harper, Peters and Gamble 2013)**.** Firstly, the platform is prone to batch effects, introduced either during bisulphite conversion or downstream processing, which may require normalization despite a strong study design. Secondly, although the two-probe biochemistry greatly increases genomic coverage, it necessitates further

within-array normalization to allow signals from both probe types to be analysed together (Dedeurwaerder *et al.* 2011). Familial data requires particular attention in choosing the most appropriate normalisation method, and this is discussed in detail in Chapter.3.

Fifty-eight samples, including fifty unique samples and eight biological replicates, as specified in Figure 2.6 (indicated by arrow heads) and Table 2.1, were subjected to methylation analysis. One microgram of DNA from each sample was bisulphite converted, using the EZ DNA Methylation-Gold (TM) kit (Zymo Research), according to the manufacturer's instructions. Bisulphite converted DNA (400ng) was then used for analysis of DNA methylation using the methylation array BeadChips, according to the manufacturer's instructions. The samples were analysed over five plates, with replicate samples analysed across the plates to allow for monitoring and adjustment of potential technical bias. Forty-eight of the fifty-eight samples selected for interrogation on the methylation array were analysed in-house, while ten were performed using a service provider, Service XS (Leiden, The Netherlands). These ten samples were analysed using a service provider as they were analysed prior to the installation of Illumina HiScan Technology within our laboratory. BeadChips analysed in-house were imaged on the Illumina HiScan reader. Upon scanning, the florescent signals are excited by a laser and Illumina's *iControl* software records these signals as *IDAT* files containing the raw intensity signals from both red and green colour channels (Illumina 2012). For each of the 480,000 loci on the array, a *beta* value between 0-1 is then generated, indicating the proportion of methylation at that site

for the population of cells analysed (0 indicative of no methylation, 1 completely methylated).

Table 2.1 Samples selected from the Tasmanian Familial Prostate Cancer Resource for methylation and genotype array analysis

	Sample ID	*Sex*	*Disease Status*	*Rel* [+]	*Quality* [M]	*Rep* [M]	*Quality* [G]	*Rep* [G]	Age *
1	PC1-03	M	Affected	1	NA	NA	Good	No	82
2	PC2-01	M	Affected	2	NA	NA	Good	No	52
3	PC2-02	M	Affected	2	NA	NA	Good	No	53
4	PC2-03	M	Affected	2	NA	NA	Good	No	58
5	PC3-01	M	Affected	3	NA	NA	Good	No	n/a
6	PC4-01	M	Affected	0	NA	NA	Good	No	74
7	PC9-01	M	Affected	2	Good	No	Good	No	64
8	PC9-04	M	Affected	0	Good	No	Good	No	65
9	PC9-12	M	Affected	0	Good	No	Good	No	72
10	PC9-121	M	Unaffected	1	Good	No	Good	No	48
11	PC9-129	F	NA	1	Good	No	Good	No	61
12	PC9-24	F	NA	1	Good	No	Good	No	45
13	PC9-286	M	Unaffected	1	Good	No	Good	No	47
14	PC9-29	F	NA	1	Good	No	Good	No	n/a
15	PC9-338	M	Affected	0	Good	No	Good	No	63
16	PC9-357	M	Unaffected	1	Good	No	Good	No	42
17	PC9-477	M	Affected	1	Good	No	Good	No	52
18	PC11-03	M	Affected	1	Good	No	Good	No	89
19	PC11-04	M	Affected	1	Good	No	Good	No	73
20	PC11-09	M	Affected	2	Good	No	Good	No	83
21	PC11-147	M	Affected	1	Good	No	Good	No	61
22	PC11-180	M	Unaffected	1	Good	No	Good	No	42
23	PC11-213	M	Affected	0	Poor	x2	NA	NA	62
24	PC11-233	M	Unaffected	3	Poor	x2	NA	NA	92
25	PC11-234	M	Unaffected	1	Good	No	Good	No	55
26	PC11-415	M	Unaffected	1	Good	No	NA	NA	61
27	PC12-04	M	Affected	2	NA	NA	Good	No	n/a
28	PC22-02	M	Affected	2	Good	No	NA	NA	64
29	PC22-03	M	Affected	2	Good	No	Good	No	74
30	PC22-04	M	Affected	1	Good	No	Good	No	62
31	PC22-16	M	Affected	3	Good	x2	Good	No	76
32	PC22-162	M	Unaffected	2	Good	No	Good	No	56
33	PC22-17	M	Affected	3	Good	x4	Good	x2	63
34	PC22-195	F	NA	1	Good	No	NA	NA	40
35	PC22-203	M	Affected	3	Good	No	Good	No	75
36	PC22-21	M	Affected	2	Good	No	Good	No	70
37	PC22-210	F	NA	4	Good	No	Good	No	73
38	PC22-274	M	Unaffected	1	Good	No	Good	No	45
39	PC22-386	F	NA	1	Good	No	NA	NA	56
40	PC22-387	M	Affected	2	Good	No	Good	No	79

41	PC22-388	M	Unaffected	3	Good	No	Good	No	73
42	PC22-393	F	NA	1	Good	No	Good	No	44
43	PC22-414	F	NA	3	Good	No	Good	No	66
44	PC22-416	M	Affected	2	Good	No	Good	No	61
45	PC22-418	M	Unaffected	3	Good	No	Good	No	54
46	PC22-468	M	Affected	0	Good	No	NA	NA	69
47	PC22-476	M	Unaffected	1	Good	No	Good	No	36
48	PC27-01	M	Affected	1	NA	NA	Good	No	64
49	PC72-02	M	Affected	1	NA	NA	Good	No	85
50	PC72-03	M	Affected	3	Good	No	Good	No	70
51	PC72-04	M	Affected	4	Good	No	Good	x2	78
52	PC72-106	M	Unaffected	2	Good	No	Good	No	46
53	PC72-126	M	Affected	1	Good	No	Good	No	49
54	PC72-136	M	Other Cancer	0	Good	No	NA	NA	57
55	PC72-187	F	NA	2	Good	No	Good	No	41
56	PC72-188	M	Other Cancer	1	Good	No	Good	No	23
57	PC72-213	M	Unaffected	0	Good	x2	NA	NA	41
58	PC72-291	M	Unaffected	1	Poor	x2	NA	NA	42
59	PC72-77	M	Affected	0	Good	No	NA	NA	75
60	PC72-94	M	Other Cancer	1	NA	NA	Good	No	61
61	PC72-97	F	Other Cancer	2	NA	NA	Good	No	n/a
62	PC75-01	M	Affected	2	NA	NA	Good	No	65

NA: Non-applicable

Rel^{+} : number of affected first degree relatives

$Quality^{M}$: quality of methylation data

Rep^{M}: analsyed as a replicate sample on the methylation array

$Quality^{G}$: quality of genotype data

Rep^{G}: analsyed as a replicate sample on the genotype array

* Age at sample collection

n/a: not available

2.2.4 Genome-wide methylation data extraction, pre-processing and initial quality control analysis

Raw data was combined from all batches generated in both the overseas and in-house laboratories, and analysed together. Illumina's *GenomeStudio* software package provides only basic quality control and pre-processing, which is insufficient to analyse familial methylation data, as this type of data requires specialised normalisation methods to remove technical bias. As such, IDAT files were analysed in the R environment (R Core Team, 2014). A combination of three R packages, *minfi* (Aryee *et al.* 2014), *methylumi* (Davis, S *et al.,* 2012) and *ChAMP* (Morris *et al.* 2014), were used to load *IDAT* files into R and perform basic quality control analysis. Different normalization methods require the data to be in different formats and it is often difficult or impossible to change the format of the data between packages once it is loaded in R. As such, a number of different packages were used to load data, with the chosen package dependent on the normalization method tested. *Methylumi* was used to read data into R in the correct format for Quantile Normalisation in the *lumi R* package.

The *minfi* package provides a quality control report based on inbuilt control probes on the array (such as staining, hybridization, bisulphite conversion and negative controls) as well as the ability to exclude probes and samples based on probe signal intensity. Samples failing this initial quality control were excluded from further analysis according to the recommendations of the *minfi* package authors. Specifically, samples were excluded if the sample profile deviated markedly from that of the other samples on both the density plot and bean plot.

Biological replicate samples were included across batches to allow assessment of quality control and technical bias. Of the 50 unique blood samples and 8 replicates initially interrogated, 47 unique and 5 replicate samples passed quality control metrics and were used for further analysis, as indicated in Figure 2.6 (green arrow heads indicating samples for which good quality data was generated and red arrow heads indicating samples which generated poor quality data) and in Table 2.1. Following sample quality control, the default quality thresholds in *ChAMP* were employed to exclude poor quality probes, with a minimum detection p-value of 0.05 in more than one sample removing 6740 probes and a bead count threshold of <3 in 5% of samples removing a further 478 probes, leaving 478,293 probes.

To account for sex differences in methylation, driven particularly by X-inactivation dosage compensation, probes on the sex chromosomes were removed prior to normalisation. While *ChAMP* includes this option as default when loading data, most packages require manual separation, normalisation and recombination of sex chromosomes or their complete manual removal. Thus to permit appropriate comparison of normalization methods, a homogenous set of loci across all packages was required and therefore sex chromosomes were removed at this stage of analysis and not re-introduced. There is debate as to whether all probes containing SNPs ought to be removed from datasets during pre-processing (Shoemaker *et al.* 2010; Chen *et al.* 2013; Zhi *et al.* 2013). This decision will vary between studies depending on the nature of the data and the research question at hand. In the current study, SNP probes were not excluded during the pre-processing, as the aim of the study is to examine the effect of genetic variation on methylation profiles. However, a subset

of SNP probes previously shown to introduce technical bias were removed during down-stream analysis, as discussed in Chapter.4 sections 4.1 & 4.3.4.

2.2.5 Genome-wide SNP genotyping of familial samples

Between 2008 and 2015 the 1000 Genomes Project (1KGP) mapped approximately 85 million single nucleotide variants across 2,500 individuals from 26 populations, down to a minor allele frequency of 1% (Auton *et al.* 2015). Drawing on this public catalogue of human variation, the Illumina Human Omni2.5 array is the first of Illumina's SNP arrays to include broad coverage of 1KGP data. Integration of data from the catalogue enabled the array to capture much of this human variation, by including 2.5 million common and rare variants down to a minor allele frequency of 2.5%.

Fifty-one unique samples were selected from the Tasmanian Familial Prostate Cancer Resource and interrogated on Illumina's Omni2.5 array. Thirty-two samples were interrogated in-house and twenty-one (including two replicate samples) were generated by the Illumina Genome Network commercial service (USA). Two samples, PC22-17 and PC72-04 were analysed in both locations to provide an opportunity to compare sample quality and examine technical bias. The thirty-two samples selected for genotyping analysis in-house were interrogated across four of Illumina's Human Omni2.5 BeadChips, hereafter referred to as the 'genotype array'. DNA at a quantity of 200 ng per sample was processed across four BeadChips according to the manufacturer's instructions. Briefly, DNA was amplified, fragmented and hybridised to BeadChips before being washed, stained and imaged on an Illumina HiScan.

Similar to the methylation BeadChips, the genotype BeadChips were imaged using an Illumina HiScan in combination with *iControl* software to generate IDAT files containing the raw intensity signals from red and green colour channels. The raw data intensity files for the combined fifty-one samples performed in-house and those analysed through the commercial service were uploaded together to Illumina's *GenomeStudio* software package.

2.2.6 Extraction, pre-processing and quality control analysis of genotype data

Raw data from both laboratories was combined and analysed in *GenomeStudio,* with initial quality control performed in accordance with Illumina's recommendations. Specifically, internal sample dependent and independent controls on the genotype array were used to visualise sample and probe quality. Sample-dependant controls included hybridisation stringency, non-specific binding and non-polymorphic controls while sample-independent controls included staining, extension, target removal, hybridisation and sample restoration.

A score indicating the quality of each genotype call (GenCall Score, ranging from 0.0 to 1.0) was calculated for each genotype using a sample-clustering algorithm. Genotype clustering at each SNP was examined by the angle, dispersion and overlap of the clusters together with signal intensity. Lower GenCall Scores were allocated to genotypes located furthest from a cluster center (See Figure 2.7 for an example of a high (A) and low (B) scoring genotype).

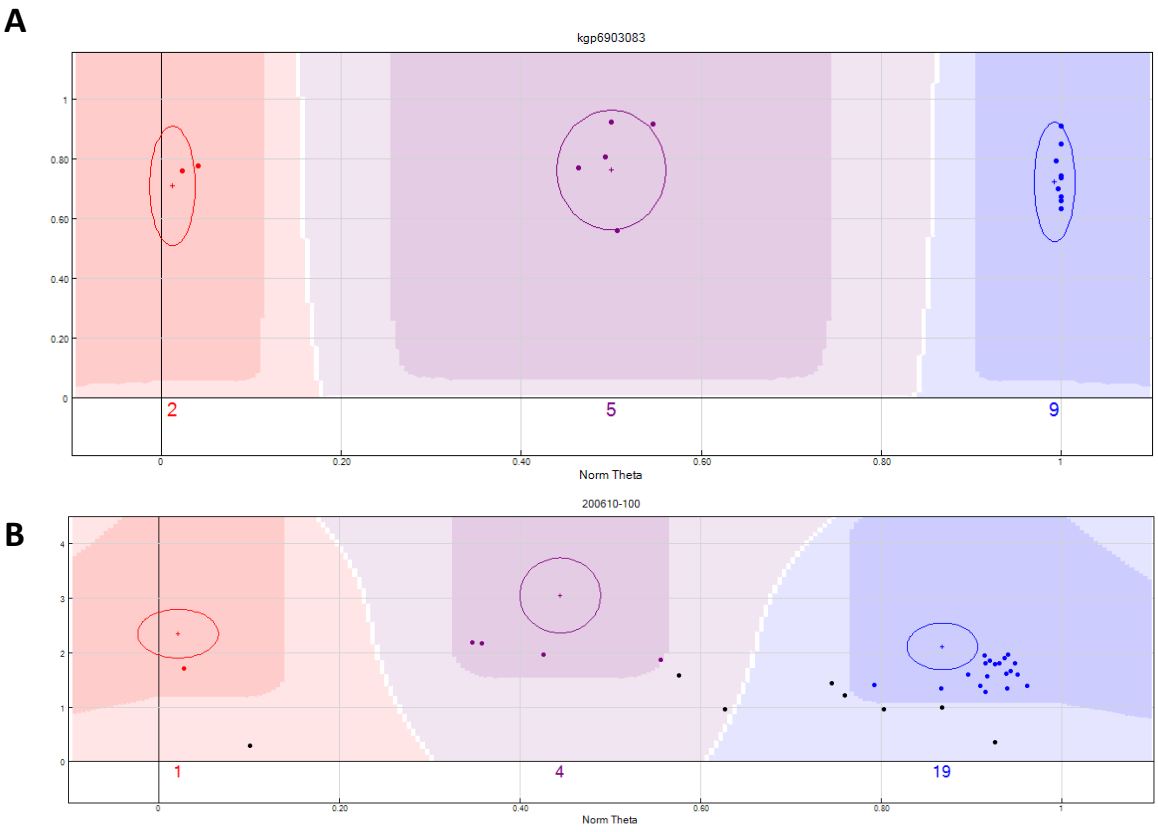


Figure 2.7 Comparison of good and poor quality SNP clusters on the genotype array.

Intensity signals for two different loci are presented. A) represents a good quality SNP call as samples are clustered in one of three tightly grouped distributions. Firstly 2 homozygous samples in pink, followed by 5 heterozygous samples in purple and finally 9 homozygous samples in blue. All samples are within or on the cluster boundaries. Contrastingly, B) represents a poorly scored locus as all samples fall outside the boundary of the cluster centre. Black samples are unable to be classified into any genotype.

Several GenCall Score thresholds were tested to find an optimal cut-off that would eliminate poorly performing SNP probes without removing probes of high quality. The default GenCall Score threshold of 0.15 was selected as it optimised the call rate without compromising reproducibility or Mendelian consistency. Using this threshold, between 2,252 and 79,204 probes were removed per sample. Table 2.2 shows aspects of the DNA quality report generated in *GenomeStudio*, detailing the number of genotype calls made at each loci, the number that were unable to be called due to poor clustering (no calls) and the call rate or proportion of good calls. The high quality of data generated across all samples on the genotype array is indicated by the high call rates for each sample, as such no samples were excluded from downstream analysis.

To further inspect sample quality and reliability, a heritability report was generated in *GenomeStudio* for a subset of samples with at least one parent present on the genotype array. Table 2.3 shows that for all thirteen samples measured, a parent-child heritability frequency of well over 99% was achieved, aligning with expected values for parent-child first degree relatives, providing further evidence of the reliability of the data.

Table 2.2 Genotype array DNA Quality Report

Number of loci = 2,391,739
Low GenCall Score Cutoff = 0.15

Row	Sample ID	No Calls	Calls	Call Rate
1	PC2-01	8389	2383350	0.997
2	PC2-02	3738	2388001	0.998
3	PC2-03	9141	2382598	0.996
4	PC22-03	3311	2388428	0.999
5	PC22-04	4275	2387464	0.998
6	PC22-16	3963	2387776	0.998
7	PC22-17_a	14596	2377143	0.994
8	PC22-21	4360	2387379	0.998
9	PC22-162	6119	2385620	0.997
10	PC22-203	6875	2384864	0.997
11	PC22-274	12465	2379274	0.995
12	PC22-387	2252	2389487	0.999
13	PC22-388	2870	2388869	0.999
14	PC72-106	6116	2385623	0.997
15	PC72-188	3577	2388162	0.999
16	PC72-02	2993	2388746	0.999
17	PC72-03	3552	2388187	0.999
18	PC72-04_a	3569	2388170	0.999
19	PC72-94	2544	2389195	0.999
20	PC72-97	4048	2387691	0.998
21	PC72-126	2417	2389322	0.999
22	PC9-477	63674	2328065	0.973
23	PC22-17_b	16957	2374782	0.993
24	PC1-03	7542	2384197	0.997
25	PC9-121	10428	2381311	0.996
26	PC11-180	57113	2334626	0.976
27	PC9-357	51992	2339747	0.978

Row	Sample ID	No Calls	Calls	Call Rate
28	PC22-418	63055	2328684	0.974
29	PC11-234	64334	2327405	0.973
30	PC9-04	35040	2356699	0.985
31	PC9-29	13796	2377943	0.994
32	PC27-01	10986	2380753	0.995
33	PC11-04	9259	2382480	0.996
34	PC22-393	6633	2385106	0.997
35	PC72-04_b	21515	2370224	0.991
36	PC22-476	5537	2386202	0.998
37	PC9-129	19829	2371910	0.992
38	PC9-01	2895	2388844	0.999
39	PC22-210	4130	2387609	0.998
40	PC12-04	39723	2352016	0.983
41	PC4-01	44277	2347462	0.982
42	PC75-01	44266	2347473	0.982
43	PC11-09	19405	2372334	0.992
44	PC9-286	19285	2372454	0.992
45	PC9-338	6608	2385131	0.997
46	PC3-01	38797	2352942	0.984
47	PC22-414	79204	2312535	0.967
48	PC11-147	53769	2337970	0.978
49	PC11-03	39124	2352615	0.984
50	PC72-187	48661	2343078	0.980
51	PC9-12	12555	2379184	0.995
52	PC22-416	56758	2334981	0.976
53	PC9-24	16873	2374866	0.993

Table 2.3 Heritability Report for a subset of genotype array data

Number of loci = 2,391,739
Low GenCall Score Cutoff = 0.15

	Child ID	Parent ID	Correct	Errors	Total	Parent-Child Heritability Freq
1	PC22-274	PC22-21	2375467	188	2375655	0.9999
2	PC72-106	PC72-04	2382249	156	2382405	0.9999
3	PC72-188	PC72-126	2385723	162	2385885	0.9999
4	PC72-94	PC72-02	2386361	155	2386516	0.9999
5	PC72-97	PC72-02	2384796	177	2384973	0.9999
6	PC72-126	PC72-04	2385891	136	2386027	0.9999
7	PC9-121	PC9-04	2347202	616	2347818	0.9997
8	PC11-180	PC11-04	2325376	2186	2327562	0.9991
9	PC9-357	PC9-338	2331699	2144	2333843	0.9991
10	PC11-234	PC11-09	2306769	2863	2309632	0.9988
11	PC22-476	PC22-416	2327717	2715	2330432	0.9988
12	PC72-187	PC72-04	2337712	2186	2339898	0.9991
13	PC9-24	PC9-12	2362774	291	2363065	0.9999

Additional sample and SNP error checking was then performed in PLINK (Purcell *et al.* 2007) on the remaining 2,391,739 SNPs. For sample quality, sex assignment was confirmed and a threshold of 10% was applied for the proportion of missing genotypes per sample. All samples passed these quality measures and progressed to the next stage of quality control, SNP error checking. Poor quality SNPs were excluded from further analysis based on three stepwise quality checks; firstly, missing genotype calls, secondly failing to reach Hardy-Weinberg equilibrium and finally minor allele frequency (MAF). As a result of a missing genotype in more than 5% of samples, 61,217 SNP probes were removed, leaving 2,330,522 loci for analysis. A Hardy-Weinberg equilibrium test was applied to the data with a threshold p-value of 0.001, with no SNPs failing this test. A MAF of 0.1% was applied to the data removing 713, 687 SNPs, leaving 1,616,835 with a total genotyping rate in remaining individuals of 0.993. Good quality data were generated for all samples analysed for genotype, as indicated by green diamonds in Figure 2.6 and as detailed in Table 2.1.

2.2.7 High quality methylation and genotype data was attained for thirty-nine samples

Of the samples analysed for both methylation and genotype, good quality data was generated for thirty-nine samples, as specified in italics in Table 2.1. These samples are also highlighted by green arrow heads and diamonds in Figure 2.6. Data for three of the replicate pairs (PC11-213, PC11-233 and PC11-191) interrogated on the methylation array failed to pass quality control metrics and were excluded from further analysis. No samples were excluded from the genotype analysis as all

samples passed minimum quality control thresholds. Overall, for the thirty-nine samples with good quality methylation and genotype data, there were eleven father-son pairs, seven brother pairs and four sibling trios, including the three-generation cluster in Family 72.

2.3 Discussion

The utility of founder populations to study complex disease genetics has been widely demonstrated, such as the identification of *BRCA1/2* as a disease susceptibility gene in the Ashkenazi Jewish population (Kirchhoff *et al.* 2004). In addition, genetic studies conducted in the Icelandic population (deCODE Genetics (Amundadottir *et al.* 2004)) and in the Mormon population in Utah (Christensen, Bonnie and George 2010) have made important contributions to our understanding of a number of complex diseases. These populations share similar features, where the current population has descended from a reduced number of founders due to the restricted population or bottleneck at some point in history. The Tasmanian population shares a number of features of these types of populations, in that the population has been relatively stable, the majority having descended from founder population originating from the UK as either convicts or settlers in the 1800s. Further, it has been demonstrated that large families can be identified and traced back to their common founders. This is exemplified by the contributions that have been made to understanding the genetic basis of disease using the Tasmanian population, including contributions to Huntington's disease (Brothers 1964), multiple endocrine

neoplasia (Burgess *et al.* 2000), glaucoma (Fingert *et al.* 1999), congenital cataract (Burdon *et al.* 2003) and most recently prostate cancer (Eeles *et al.* 2009).

The utlilisation of the Tasmanian Familial Prostate Cancer Resource here, is particularly pertinent to highly penetrant disease variants as the original affected men were recruited to the study after presenting with symptomatic disease. This is in contrast to other prostate cancer studies that were initiated subsequent to wide-spread introduction of PSA testing which faced the widely recognized issue that PSA-detected disease does not necessarily equate to clinicaly relevant prostate cancer (Saini 2016). Additionally, the Tasmanian resource now contains genealogical and pathology records for over one thousand people from sixty large families, often extending over five or six generations. Thus, this resource provides a rich source from which to draw genomic and epigenomic data to help understand the underlying mechanisms contributing to prostate cancer predisposition.

Familial genetic resources such as the one used here are relatively rare, particularly in prostate cancer where disease onset is late and thus multiple generational families are rare. However, there are a few comparable resources for examining prostate cancer risk, comprising very large multiple affected generation families which can be traced back to their founders. Rich genealogical data is available in Iceland (deCODE (Amundadottir *et al.* 2004)) and Utah populations (Christensen, Bonnie and George 2010)and this has facilitated the generation of valuable genetic resources for complex disease research including studies into prostate cancer. Other

familial resources exist in Finland, Sweden and the US (Seattle, Washington State) (FitzGerald *et al.* 2013).

While these types of familial resources have the potential to provide invaluable insight into disease processes, samples must be carefully selected from extensive pedigrees to provide the most informative data (Ziegler and Sun 2012). Additionally, there are several limitations associated with familial study design. Samples may not be available for all individuals of interest, as samples are frequently collected over extended time periods, often spanning many decades. This long acquisition time can also lead to degradation of sample material. Additionally, there is often a limited quantity of genomic material available, especially if many family members in extended pedigrees are deceased, as is often the case in generations deceased prior to the pedigree being ascertained. There were three main phases throughout the design of this study where samples were excluded due to poor quality. Initially, samples with insufficient genomic material or inappropriate quality were excluded. Secondly, data generated for each sample through methylation and genotype analysis were required to pass quality control thresholds for each individual platform, as described in sections 2.2.4 and 2.2.6. A further limitation of this study was the removal of probes located on the sex chromosomes. These probes were excluded for consistency as not every R package used in the methylation array analysis allowed for separate normalisation of sex chromosomes and autosomes. This procedure is imperative as X-inactivation dosage compensation leads to high methylation levels on the X chromosome. Finally, samples were only able to be used to examine the association between genotype and methylation if data of high quality

was generated across both platforms. Thirty-nine samples passed both quality control thresholds and were thus used for this subsequent analysis, as described in Chapter 4.

Despite these limitations, familial data sets present an exciting opportunity to examine rare deleterious variants (Williams and Blangero 1999). This chapter has described the rationale used to prioritise samples from the Tasmanian Familial Prostate Cancer Resource for use in this study, as well as detailing the creation of high quality genome-wide genotype and epigenomic data for subsequent analysis.

Chapter 3 – Development of appropriate normalisation strategies for the analysis of germline familial methylation data

3.1 Introduction

To date, genome-wide epigenetic studies have largely focused on epigenetic alterations that occur in diseased tissues, where epigenetic changes across the genome are mapped through comparing 'normal' and affected tissues from the same individual. Indeed epigenetic drugs, currently in clinical use are designed to correct the epigenetic alterations acquired during disease development (Sharma, Kelly and Jones 2010); the assumption being that these acquired epigenetic alterations are driven by the disease process itself. More recently it has been hypothesised that inherited genetic variation can drive epigenetic alterations and further that these contribute to disease susceptibility or disease course. To date the large majority of genome-wide methylation studies and consequently the bioinformatics pipelines used to interpret these data have been designed to compare diseased tissue with 'normal' tissue, in order to map epigenetic changes in the diseased tissue itself. This analysis necessarily screens out inherited epigenetic changes that are evident both in the normal tissue and the diseased tissue of the same affected individual. There remains a need to explore inter-individual variation of the epigenome and it's contribution to disease, as evidence suggests genotype is one of the greatest influences on epigenetic patterns (Gertz *et al.* 2011). For a more

detailed description on the inherited drivers of methylation patterns see section 1.5.4.1 in Chapter 1.

A powerful approach for examining the role of inherited variation driving epigenetic change, is to examine large families where inheritance and methylation patterns can be tracked through generations. A number of challenges exist in the analysis of genome-wide methylation mapping in samples and these include technical challenges dealing with batch effects and the underlying biochemistry employed by the array methods. This has necessitated the development of numerous pre-processing quality control methods to ensure reliable, high quality data generation. As mentioned above, most studies examining epigenetic profiles typically assess differences between two distinct groups (normal tissue versus tumour tissue or case vs control status), and as such the majority of normalisation methods for the analysis of methylation array data are designed for these types of comparisons. These methods frequently require two data groups to normalise negative and positive control probes or genomic regions. Such methods are incompatible with pedigree data, which lack a distinct second comparison group for normalisation. While other methods such as *BMIQ* (Teschendorff *et al.* 2013) do not rely on cancer-normal differences, they often focus on intra-array normalisation and do not necessarily address technical bias between samples. To avoid unwanted technical noise such as batch affects, it may therefore be necessary to combine such intra-array normalisation methods with an inter-array correction method such as *ComBat* from the *sva* R package (Leek *et al.* 2012). In the absence of appropriate strategies for pre-processing familial-based methylation array data, there is a need to develop a 'fit for

purpose' pipeline, which can be used as a guide to remove poor quality samples and probe signals, adjust for technical bias and prepare data for analysis of biological differences. As such, the aim of this chapter is to establish and test a pipeline to remove technical bias whilst maintaining biological information in familial data generated on the methylation array.

3.2 Method

3.2.1 Normalisation

To determine the best normalisation method for familial data, eight techniques, as described in Table 3.1, were applied to familial clusters of samples selected from the larger pedigrees, as detailed in Figure 2.6 of Chapter 2. Section 2.2.1 of Chapter 2 contains a detailed description of the laboratory processing of the methylation array and initial quality control.

Table 3.1 Normalisation methods tested. The table includes a brief description of each method, the relevant R package and reference for further information.

Normalisation method	Package	Reference
Quantile Normalisation The distributions of probe intensities for different samples are made identical. Often used in microarray analysis.	lumi	(Du, Kibbe and Lin 2008)
Stratified Quantile Normalisation Probes are stratified by genomic region then quantile normalised with sex chromosomes normalised separately when male and female samples are present. No background correction, zeros removed by outlier function. Not recommended for cancer-normal comparisons or other groups with global differences.	minfi	(Aryee *et al.* 2014)
Beta-Mixture Quantile Dilation (BMIQ) Adjusts type II probes to type I distribution. Recommended for all datasets.	ChAMP	(Teschendorff *et al.* 2013)
Subset-quantile Within Array Normalisation (SWAN) A quantile distribution is created using a subset of probes, with subsetting based on the number of CpGs in the probe body. Separate subsets are created for type I and II probes. The remaining probes are then adjusted to the subsets.	minfi	(Maksimovic, Gordon and Oshlack 2012)
Functional Normalisation (FunNorm) Uses control probes to remove unwanted technical variation. Also diminishes batch effects in some datasets. Suitable for use in cancer-normal studies or where global methylation changes occur.	minfi	(Fortin *et al.* 2014)
Dasen Background adjustment and between-array normalisation are performed separately on type I and II probes.	wateR-melon	(Pidsley *et al.* 2013)
Noob Uses type I probe design to measure non-specific fluorescence in opposite colour channel.	minfi	(Triche *et al.* 2013)
Remove Unwanted Variation (RUV) Previously used with microarray data to normalise via negative control genes. Requires distinct groups such as cancer-normal to normalise on.	RUV-normalize	(Gagnon-Bartsch, Jacob and Speed 2012)
Batch Correction: *ComBat* Adjusts for known or unknown batches using an empirical Bayesian framework.	sva	(Leek *et al.* 2012)

The probe sub-set chosen for each analysis was selected following the instructions of each individual normalisation package, which had different requirements. This dictated whether normalisation methods were compatible and could be used in conjunction. The methods involve various degrees of type I and II probe scaling to account for underlying technical differences between the probe types, background and dye bias correction and initial batch correction between arrays. Depending on the normalisation method, data was either used in the red/green signal format (RGset), converted into methylated and unmethylated values (MethylSet) or converted to β values by the function $\beta = M/(M + U + 100)$, where M is the methylated signal and U, unmethylated. In some normalisation methods, the offset of 100 is included to regularize scores when both methylated and unmethylated values are very low. While the β value is more biologically intuitive (it ranges from 0-1 indicating the proportion of methylation at that site for the population of cells analysed), it suffers from severe heteroskedasticity at very high or low values (Du *et al.* 2010). Logit transforming to an M-value removes this unequal variance. Thus wherever possible, calculations in this study have been performed on the M-values and transformed back to β values if required for biological interpretation. Eight performance metrics were then used to compare methods and determine the optimal normalisation approach for familial datasets. Visual tools such as density and MDS plots and unsupervised hierarchical clustering were used to compare the various methods between all samples and particularly replicate samples. See Table 3.2 for a description of each metric employed to assess performance of normalisation methods.

Table 3.2 Qualitative and Quantitative metrics used to assess normalisation efficacy. The table includes a brief description of each metric and which figures describe the results for that method.

	Method	Description	Figure
1	Density Plot: all samples	Bimodal distribution of Beta values as methylated and unmethylated signals. Each sample is represented by a single line. A batch effect is indicated when samples performed in the same batch have a similar distribution.	Figure 3.2 (A,C,E); Figure 3.3
	Density Plot: 3 groups of replicate samples	Bimodal distribution of Beta values as methylated and unmethylated signals. Samples are coloured by replicate group. As each replicate group contain the same biological information, differences in sample distribution within groups indicate technical bias.	Figure 3.4 (A,C,E)
	Density Plot: Probe I and II distribution	Bimodal distribution of Beta values as methylated and unmethylated signals separated by Infinium I and II probe types. Provides information about probe normalisation which is required for Infinium I and II signals to be combined in the same analysis.	Figure 3.2 (B, D, F)
2	MDS plot: all samples	Multi-dimensional scaling plots show a 2-d projection of distances between samples. For these plots the 1000 most variable sites have been selected as they are the most biologically relevant for this type of analysis. Samples cluster by similarity and as such batch effects and familial clustering can be clearly discerned.	Figure 3.1; Figure 3.5
	MDS plot: 3 groups of replicate samples	1000 most variable sites are again selected, with samples coloured by replicate group. As each replicate group contains the same biological information, close within group clustering indicates minimal technical bias while distantly clustered replicate samples indicate heightened technical bias.	Figure 3.4 (B,D,F)
3	ANOVA of the first principal component for MDS plots	Provides a quantitative value for MDS plots. A lower p-value indicates the clustering is more significantly explained by batch, with. a larger p-value after normalisation indicating a reduction in batch effect.	P-values displayed on Figure 3.1

4	Median Absolute Differences between replicate samples	For each replicate group the median M value (log of Beta values) across all probes was calculated and the absolute difference compared between replicate groups after various normalisation methods. A smaller absolute difference indicates improved normalisation as more techincal bias is removed.	Table 3.3
5	Imprinted regions: Density plots	227 probes mapping known imprinted hemi-methylated regions can be used as a standard to measure changes in methylation levels after normalisation. Density plots have a single distribution peak since there is roughly 50% methylation at these sites.	Figure 3.7
	Differentially Methylated Region Standard Error (DMRSE)	The DMRSE measures how each sample varies from the expected 50% methylation. Smaller error/deviation from 50% indicates less technical bias.	Table 3.4; Figure 3.7 (A,C,E)
6	Cluster Dendorgram	Another tool to measure clustering by sample similarity. Samples are labelled by batch with batch effects clearly seen before normalisation and diminished after. Red stars indicate replicate samples that are expected to cluster most closely.	Figure 3.6
7	meQTL Association	Association between methylation at cg17749961 and SNPs in a 2Mb window. A significant association is maintained after normalisation and batch correction.	Figure 3.8

3.2.2 Batch Correction

Since an obvious batch effect remained after normalisation, the ComBat function from the *sva* package (Frommer *et al.* 1992) was used to further remove technical bias introduced by interrogating samples on the methylation array in different batches.

3.2.3 Statistical Analysis

Eight methods, as described in Table 3.2, were used to compare the efficacy of the various normalisation methods. In addition to density and MDS plots, the ANOVA test and quantitative measures, mean absolute difference between replicates and the differentially methylated region standard error (DMRSE) measures were used. Additionally, two approaches were taken to test the underlying biological information was preserved between samples; namely, an association analysis between genotype and methylation at a previously identified meQTL and an epigenome-wide association analysis with age.

For a qualitative measure to examine effectiveness of between array normalisation, hierarchical cluster dendrograms were generated using all probes with the *hclust* function using the Euclidean distance method from the default R package, *stats*. Cluster dendrograms group samples by differences, with similar samples grouping together.

MDS plots were clustered by batch or family, then analysis of variance was performed on the first principal component from a principal component analysis

(PCA) on the 1000 most variable beta values using the *aov* and *prcomp* functions in the *stats* core R package. P-values are displayed on the MDS plots in Figure 3.1 A lower p-value indicates clustering is more significantly explained by batch or family, with a larger p-value after normalisation indicating a reduction in technical bias.

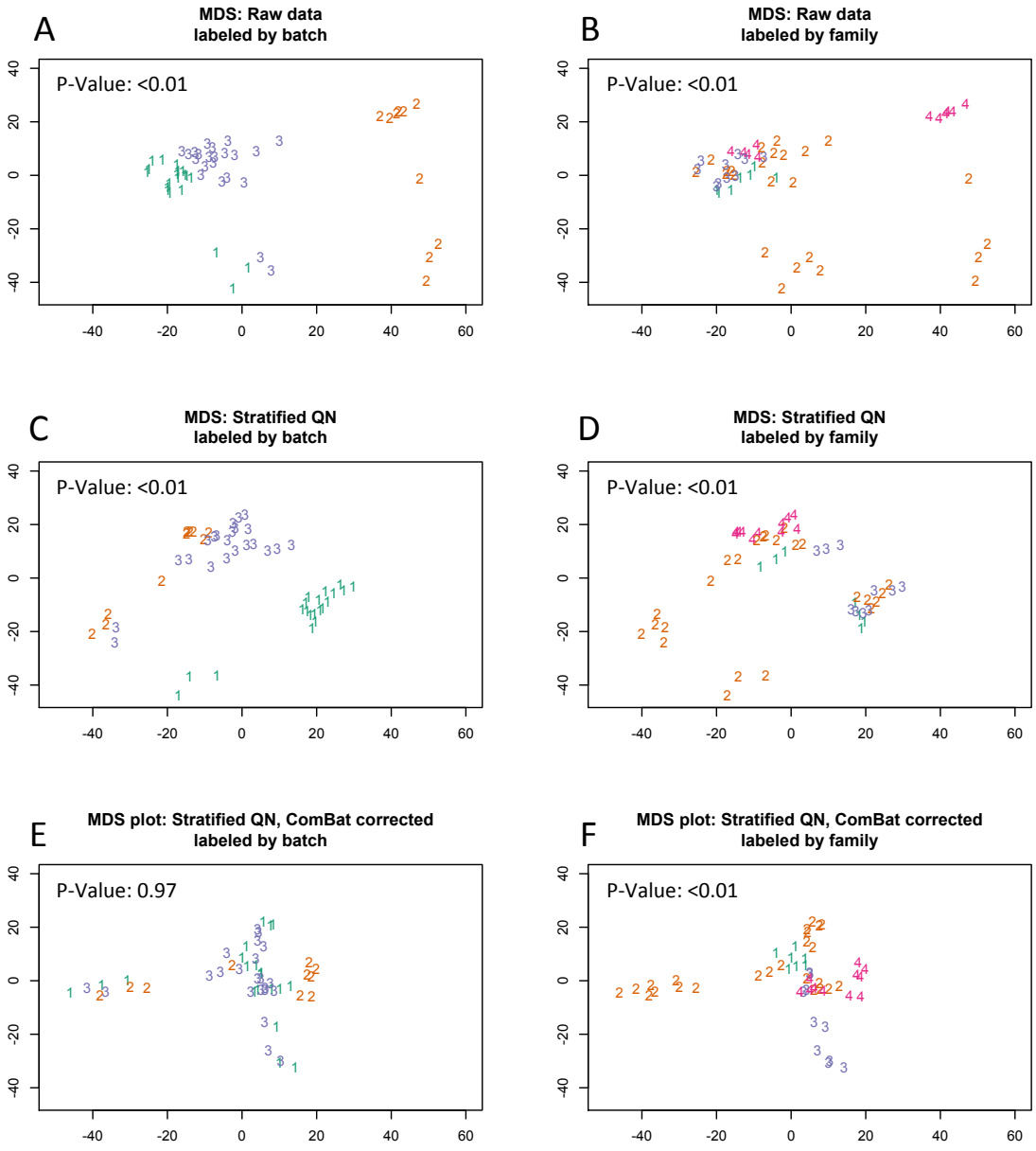


Figure 3.1 Multidimensional scaling plots of M-values by batch and family. Multidimensional scaling plots for Raw (A, B), Stratified QN (C, D) and ComBat Stratified QN (E, F) M-values. For each plot the 1000 most variable probes were selected. In A, C and E numbers represent batches and are coloured accordingly, with clustering by batch clearly seen in A, to a lesser extent in C and removed in E. In B, D and F numbers represent family groups and are coloured accordingly with the clearest clustering present in F after the batch effect has been removed.

Six replicate sample pairs were used to quantitatively assess the performance of the normalisation methods, as one sample from each pair was interrogated on a separate batch. The median absolute difference between each pair was calculated by first taking the absolute difference at each probe between the two replicates and then taking the median of the differences. A lower median difference indicates less technical bias, as the samples are biologically identical.

There are 227 known imprinted regions (iDMRs) on the methylation array, and these have previously been employed in analysis packages such as *wateRmelon* as a quality control metric (Pidsley *et al.* 2013). These regions are expected to have allele-specific methylation and a β value of 0.5, and therefore deviation from this value can be examined as a standard error-type measure, denoted DMRSE in the *wateRmelon* package. The *dmrse_row* function was used to measure dispersion of methylation between samples for each normalisation method. A lower value indicates methylation values are more tightly aligned with expected methylation levels.

Whilst evidence of clustering according to familial relationships following normalisation correction provides some confidence that biological integrity of the data is preserved, to further test the preservation of biologically relevant information, we examined detectable associations of known meQTLs in our data. Shoemaker and colleagues have previously identified 736 CpG sites to be associated with SNPs in *cis* (Zhang *et al.* 2010). Here, cg17749961, one of the ten most significant hits reported by Shoemaker et al., was examined in a subset (22 males) of the 39 individuals, for whom both methylation and genotyping SNP data was

available. Association analysis was performed between this probe site and SNPs located within a 2Mb window adjacent to this site, using linear regression, and assuming an additive disease model. Relatedness was adjusted for by fitting a linear mixed model on the methylation of cg17749961 and a kinship matrix, determined by the Identity-by-State function in the *GenABEL* R package (*GenABEL* project developers, 2013). The residuals from this model were then used as the outcome variable in the linear regression model with SNPs drawn from a 370K Illumina array. Bonferroni correction was used to correct for multiple testing error.

3.3 Results

In the previous chapter, high quality methylation data was generated for forty-seven unique and five replicate samples from four families using the methylation array in three separate batches (see Chapter 2 Table 2.1 for a list of these samples). The analysis described here has endeavoured to determine the most suitable method of removing technical bias in familial studies, by evaluating eight normalisation metrics. Both qualitative and quantitative analysis was used to examine these methods and furthermore, the data was tested to determine if true biological associations could still be identified.

3.3.1 Evaluation of normalisation methods to address technical bias

Data generated from whole genome methylation analysis employing array technology generates an output necessitating application of normalisation methods to correct for possible bias arising from within and between array variation. Herein

eight different methodologies (Table 3.1) were examined and visual and quantitative metrics were employed to evaluate their comparative performance. A minimum of one sample in each of the three batches was replicated, providing five technical replicates in addition to the three unique samples on each batch, to permit generation of data from analysis of the same biological sample. In data lacking technical bias, replicate samples would be expected to generate the most similar methylation profiles, while methylation profiles generated from closely related individuals should also cluster tightly compared to distantly or unrelated individuals. However, if technical bias such as a batch effect has been introduced, this distorts the profiles and samples no longer cluster by biological similarity but instead the most evident grouping would be by batch.

Batch effect (between array variation) was initially examined using density distribution plots (Figure 3.2). Figure 3.2A comprising raw values from all 3 batches reveals significant bias. The greatest contributor to batch effect was the date on which the BeadChips were processed, with bisulphite conversion performed on the same day as BeadChip processing. Employing a stratified QN (Figure 3.2C) and/or ComBat normalisation (Figure 3.2E) dramatically reduced this observed effect. For between array biases, Figure 3.2 shows the density distribution of β values for raw data samples (A), after Stratified QN (C) and after Stratified QN combined with ComBat correction (E). This is particularly evident when comparing the B value density plots of 3 groups of replicate samples (Figure 3.4 A, C and E).

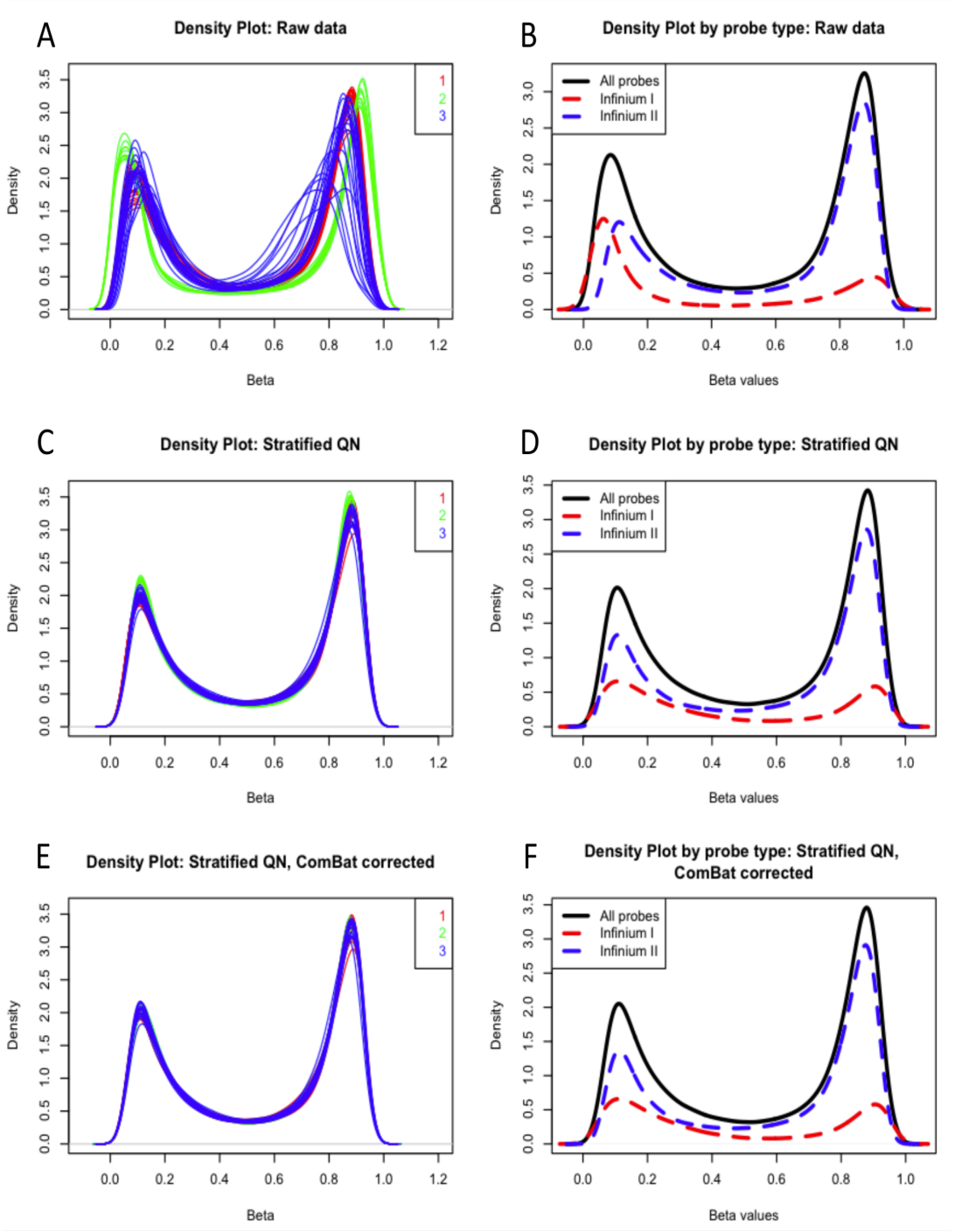


Figure 3.2 Density distribution of β values. Density plot and probe distribution of β values for Raw pre-normalisation data (A, B), after Stratified QN (C, D) and with Stratified QN and ComBat batch correction (E, F). For density plots (A, C, E) a single line represents a sample, with samples coloured by batch. A clear batch effect is present in A, lessened in C and removed in E. For the probe distribution (B, D, F) one sample has been chosen with the red dashed line indicating type I probe distribution, the blue dashed line type II and the solid black line the combined probe distribution. The probe type distribution is also improved after normalisation, as type I and II are more closely aligned in D and F compared to B.

Stratified QN also performs best at removing within array biases as the distribution of probe I and II types become more uniform (Figure 3.2 B, D and F). This bias is driven by the differing biochemistry of the probes, with type I employing a single colour channel with a different bead for methylated and unmethylated DNA and type II containing one bead in two colour channels. The underlying biology targeted by each probe is confounded by this technical bias, as type I measures CpG dense regions (such as islands) while type II can only tolerate three CpGs in the length of the probe. As such, type I interrogate a greater proportion of unmethylated to methylated DNA, while type II perform the opposite. Removing the probe bias is imperative for accurate comparisons between these probe types when pooling probe I and II data, which is necessary for accurate genome-wide methylation information of both CpG rich and poor regions.

In contrast, the density plots of β values for other normalisation (SWAN and FunNorm) methods do not improve to the same degree and in some cases greater variation is introduced (Figure 3.3C-G). For example, a worsening of the batch effect is seen for SWAN normalisation (Figure 3.3D), compared to raw data (Figure 3.3A) and the distribution of methylated and unmethylated signals is inverted following FunNorm (Figure 3.3E).

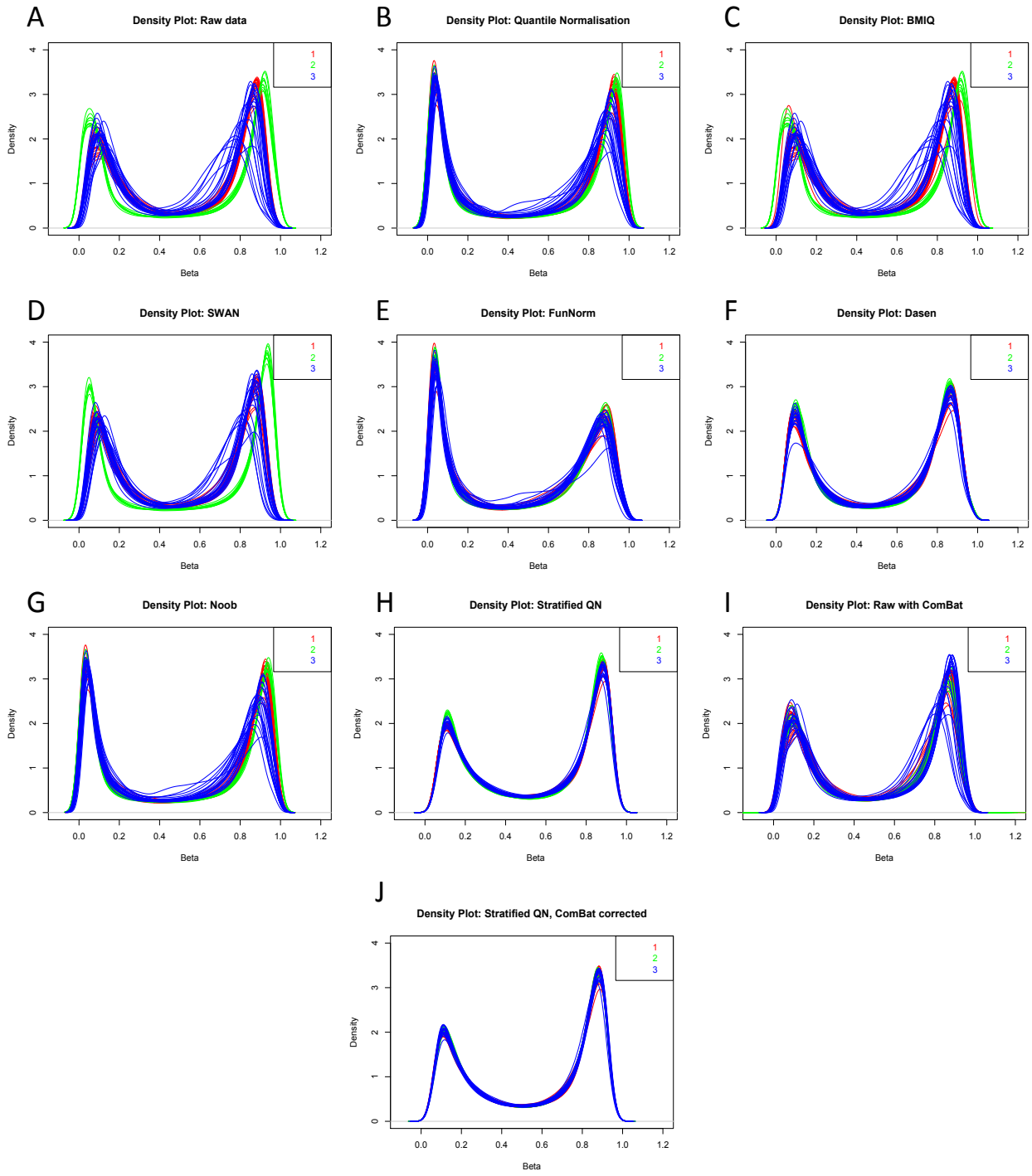

A
Density Plot: Raw data
B
Density Plot: Quantile Normalisation
C
Density Plot: BMIQ
D
Density Plot: SWAN
E
Density Plot: FunNorm
F
Density Plot: Dasen
G
Density Plot: Noob
H
Density Plot: Stratified QN
I
Density Plot: Raw with ComBat
J
Density Plot: Stratified QN, ComBat corrected
Density
Beta
1
2
3

Figure 3.3 Density distribution of β values for all normalisation methods. Density plots of β values for various normalisation methods: raw pre-normalisation data (A), Quantile Normalisation (B), BMIQ (C), SWAN (D), FunNorm (E), Dasen (F), Noob (G), Stratified QN (H), Raw with ComBat correction (I) and Stratified QN with ComBat correction (J). A single line represents a sample with samples coloured by batch. The batch effect present in the raw data (A) remains after the majority of normalisation methods with Dasen (F) and Stratified QN (H) showing the most uniform distributions. Some methods such as quantile normalisation (B) and FunNorm (E) flip the methylated and unmethylated signal distribution. ComBat is effective at removing batch effects in both raw (I) and normalised (J) data, with the best outcome seen with Stratified QN with ComBat batch correction (J).

The second approach employed to examine the performance of the normalisation methods was to generate multi-dimensional scaling (MDS) plots. These permitted the visualization of the two-dimensional projection of the differences between samples. For each plot the 1000 most variable probes were selected, as these represent the most pertinent biological differences between samples. M-values were used as opposed to β–values, the latter of which have been shown to suffer severe heteroskedasticity at very high and low values (Du *et al.* 2010). Again a strong batch effect is observed in the raw data (Figure 3.1A) as expected and this is removed or significantly reduced following normalisation using Stratified QN (Figure 3.1C) and Combat (Figure 3.1E) corrected data. The strong batch effect masks the familial relationships in the raw data, however following correction, clustering according to kinship is clearly evident (Figure 3.1F). Similarly the replicate samples (Figure 3.4) which group disparately in the raw data (A,B), co-locate or cluster tightly following Stratified QN (C,D) and ComBat (E,F). The MDS plots for each normalisation method (Figure 3.5) also show Stratified QN followed by ComBat to be the most effective method for removing clustering by batch.

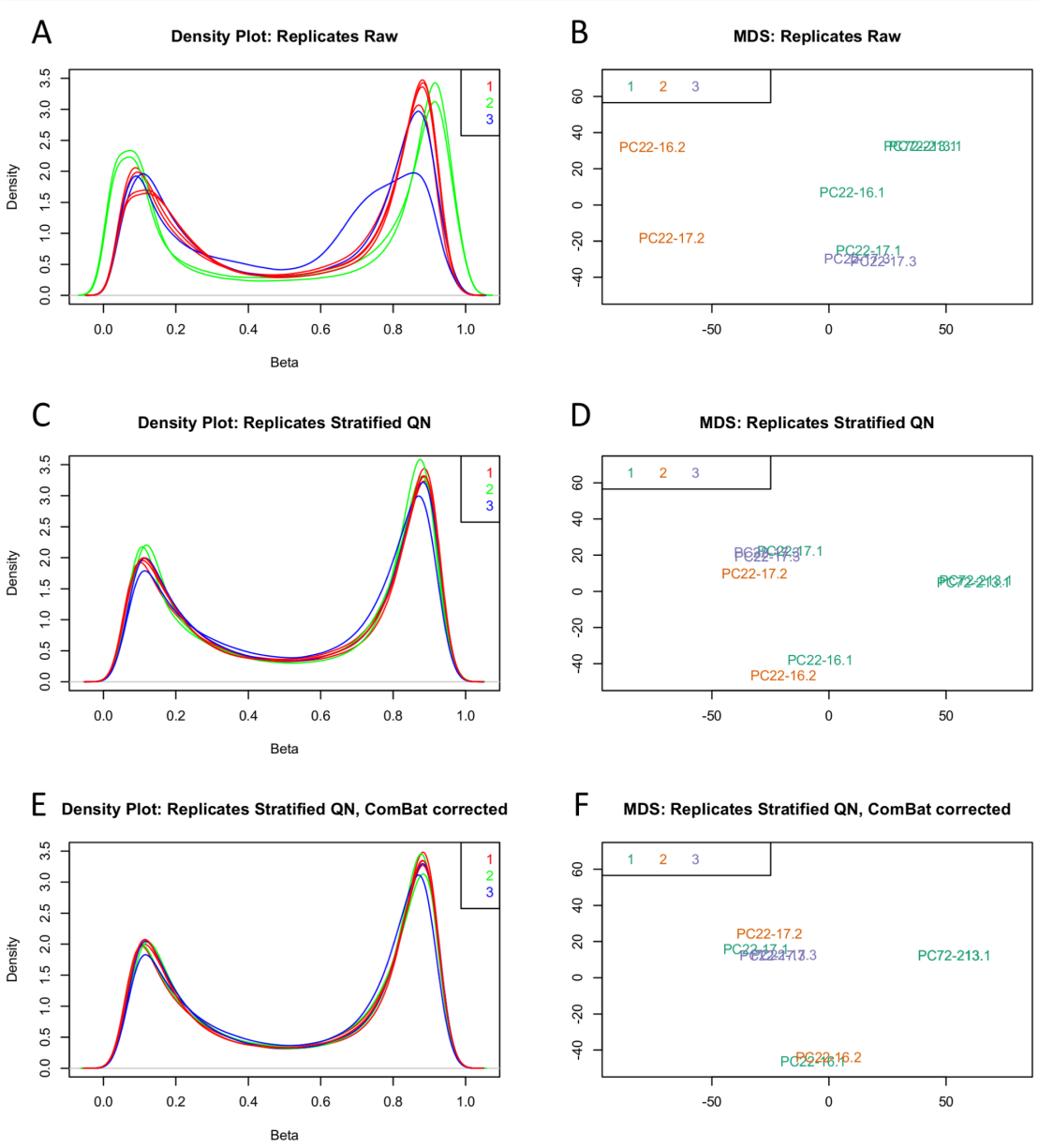


Figure 3.4 Density distribution of β values and Multidimensional scaling plots of M-values for replicate samples.

Density (A, C, E) and MDS (B, D, F) plots of three replicate sample groups for Raw (A, B), Stratified QN (C, D) and Stratified QN ComBat corrected (E, F) data. For all plots samples are coloured by batch 1-3 as labelled. Density plots show the distribution of β values, which become more uniform after Stratified QN (C) and Stratified QN plus Combat (E). MDS plots show clustering of the 1000 most variable sites by M-value, highlighting the decreasing variance between replicate groups after Stratified QN and Combat (F).

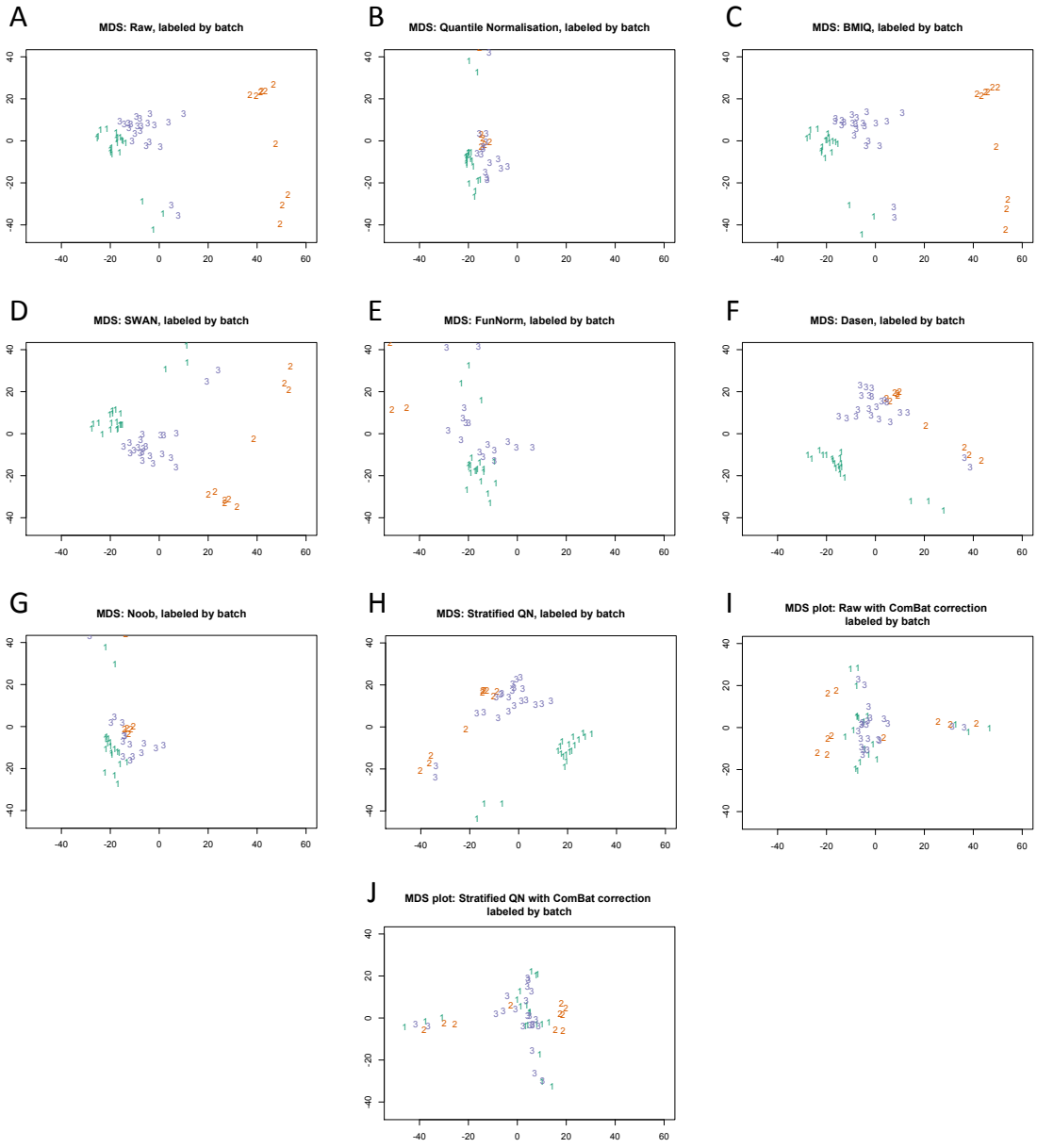


Figure 3.5 Multidimensional scaling plots of M-values by batch for all normalisation methods. Multidimensional scaling plots for Raw (A), Quantile Normalisation (B), BMIQ (C), SWAN (D), FunNorm (E), Dasen (F), Noob (G), Stratified QN (H), Raw with ComBat correction (I) and Stratified QN with ComBat correction (J). For each plot the 1000 most variable probes were selected. Batches are numbered and coloured, with clustering by batch clearly seen in the raw data (A) and removed to varying degrees with different normalisation methods. ComBat correction following Stratified QN provides optimal batch correction removal as the samples no longer cluster according to batch.

This efficacy of normalisation methods in reducing clustering of samples by batch was assessed quantitatively by ANOVA to test the effect of batch on the first principal component. The ANOVA was repeated for each normalisation method, using M-values from the top 1000 most variable sites. Consistent with the visualized MDS plot, the p-value was highly significant demonstrating the significant association of batch in M value in raw and Stratified QN data ($P<0.01$) but was not significant following correction using ComBat ($p=0.97$).

For a final qualitative measure to examine effectiveness of between array normalisation, hierarchical cluster dendrograms were generated. Application of Stratified QN and ComBat (Figure 3.6), again demonstrated superior normalisation when visualized by this method, with raw data samples clearly clustering into three distinct groups (Figure 3.6A), stratified QN resulting in improved clustering (B) while ComBat batch correction following Stratified QN completely removes the batch effect (C) permitting the desired outcome with related individuals clustering together in familial groups. Furthermore, replicate samples cluster more clearly after ComBat normalisation (C, red stars) indicating removal of batch effects without perturbing biologically relevant information.

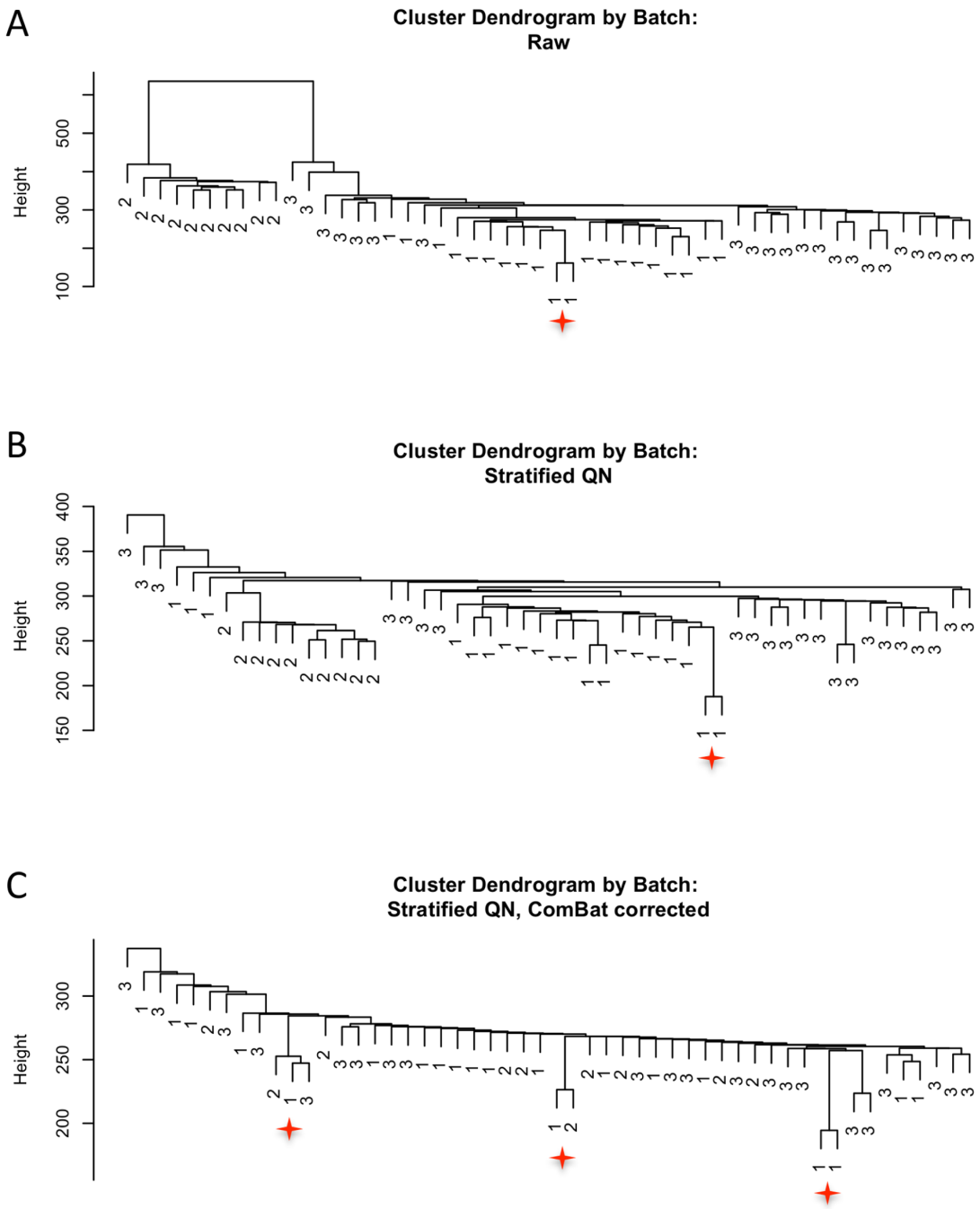


Figure 3.6 Hierarchical cluster dendrogram for Raw, Stratified QN and Combat corrected data. Samples are clustered by similarity and labelled by batch. Raw data samples (A) clearly cluster into three distinct batches while Stratified QN (B) partially adjusts clustering by batch and Stratified QN combined with Combat considerably diminishes the batch effect (C). Red stars indicate replicate samples which cluster more clearly in (C), indicating removal of batch effects.

To quantitatively assess the performance of these normalisation methods, the median absolute difference in M-values was calculated for 6 replicate pairs, with one sample from each pair interrogated on a separate batch. With the exception of one pair, Stratified QN with ComBat was found to have the lowest absolute median difference between technical replicate pairs, corresponding to the highest correlation between replicate pairs (see Table 3.3). Whilst others such as SWAN introduced an increase in the error rate relative to the Raw data values.

Table 3.3 Median absolute difference between technical replicate pairs.

Normalisation Method	**Pair.1**	**Pair.2**	**Pair.3**	**Pair.4**	**Pair.5**	**Pair.6**
Raw	0.557	0.397	0.639	0.255	0.974	0.721
Quantile Normalisation	0.335	0.612	0.378	0.322	0.610	0.414
Stratified Quantile Normalisation	0.258	0.377	0.309	0.268	0.381	0.330
BMIQ	0.569	0.414	0.646	0.271	0.980	0.726
SWAN	0.676	0.375	0.751	0.247	1.003	0.808
Functional Normalisation	0.334	0.511	0.378	0.312	0.590	0.398
Dasen	0.250	0.399	0.290	0.253	0.399	0.313
Noob	0.414	0.646	0.410	0.411	0.916	0.621
Raw with ComBat	0.263	0.268	0.258	0.236	0.336	0.261
Stratified Quantile Normalisation with ComBat	0.193	0.313	0.218	0.210	0.270	0.223

Finally, standard error measures for imprinted regions were calculated and compared between methods as described in the statistical analysis section of the methods. Smaller values indicate lower errors and more reliable data. A standard error measure (DMRSE) of 0.0048 was calculated for the raw data, with this value increasing with following normalisations using QN (0.0052), Noob (0.0052) and Functional Normalisation (0.0056). The remaining normalisation methods generated reduced DMSRE values with Stratified QN with ComBat batch correction again producing the smallest error values at 0.0012. See Table 3.4 for a full list of DMRSE values and Figure 3.7 for the density plots of these probes.

Table 3.4 Standard error measures for imprinted differentially methylated regions for the various normalisation methods.

Normalisation Method	**DMRSE ***
Raw	0.0048
Quantile Normalisation	0.0052
Stratified Quantile Normalisation	0.0028
BMIQ	0.0048
SWAN	0.0046
Functional Normalisation	0.0056
Dasen	0.0043
Noob	0.0052
Raw with ComBat	0.0028
Stratified Quantile Normalisation + ComBat	0.0012

*Differentially Methylated Region Standard Erro

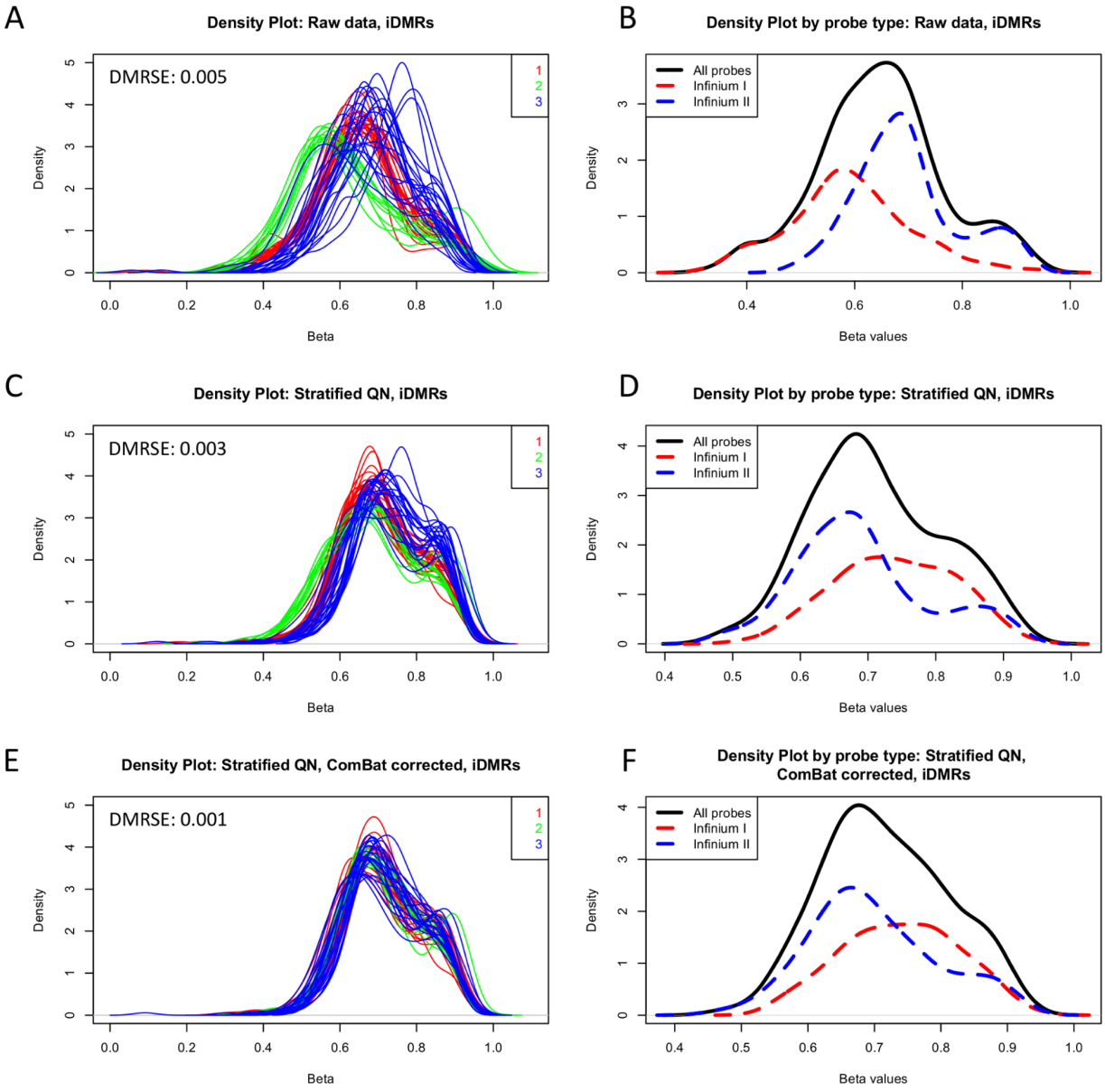


Figure 3.7 Density distribution of β values for imprinted differentially methylated regions. Density plots for Raw (A), Stratified QN (C) and Stratified QN with ComBat (E) for 227 probes mapping known imprinted differentially methylated regions. Each line represents a sample, with samples coloured by batch. As methylation at these loci is allele-specific there is a single density distribution rather than the bimodal distribution seen in Figures 3.2, 3.3 and 3.4A,C,E. The standard error type measure (DMRSE) diminishes with Stratified QN and Combat, indicating more reliable data. B, D and F show the Infinium I and II probe distributions, which becomes more uniform with Stratified QN and Combat.

3.3.2 Increased power for determining true biological associations

Critical to any normalisation method is the maintenance of true biological differences between samples. As described in the statistical analysis section of the methods, a previously identified meQTL (cg17749961, (Zhang *et al.* 2010)) was selected to perform association analysis with prior to and following normalisation. Following Bonferoni correction, a significant association was detected in the raw data (Figure 3.8A, p-value=7.29e-06), increasing markedly after Stratified QN (Figure 3.8B, p-value=3.53e-07). After ComBat (C) there was a drop in significance compared to Stratified QN and Raw data, yet the p-value was still highly significant (p-value=1.05e-05) indicating preservation of the biological information of interest. The drop in significance after batch correction may be explained as confounding between batch and family, which is removed after ComBat. Ideally, samples would be randomised across experiments, however the nature of familial studies is such that this is not always possible, as samples are collected at different time points, often across generations. To maintain maximum power, the inclusion of all available samples is essential and therefore, data processing methods capable of dealing with non-ideal datasets are required.

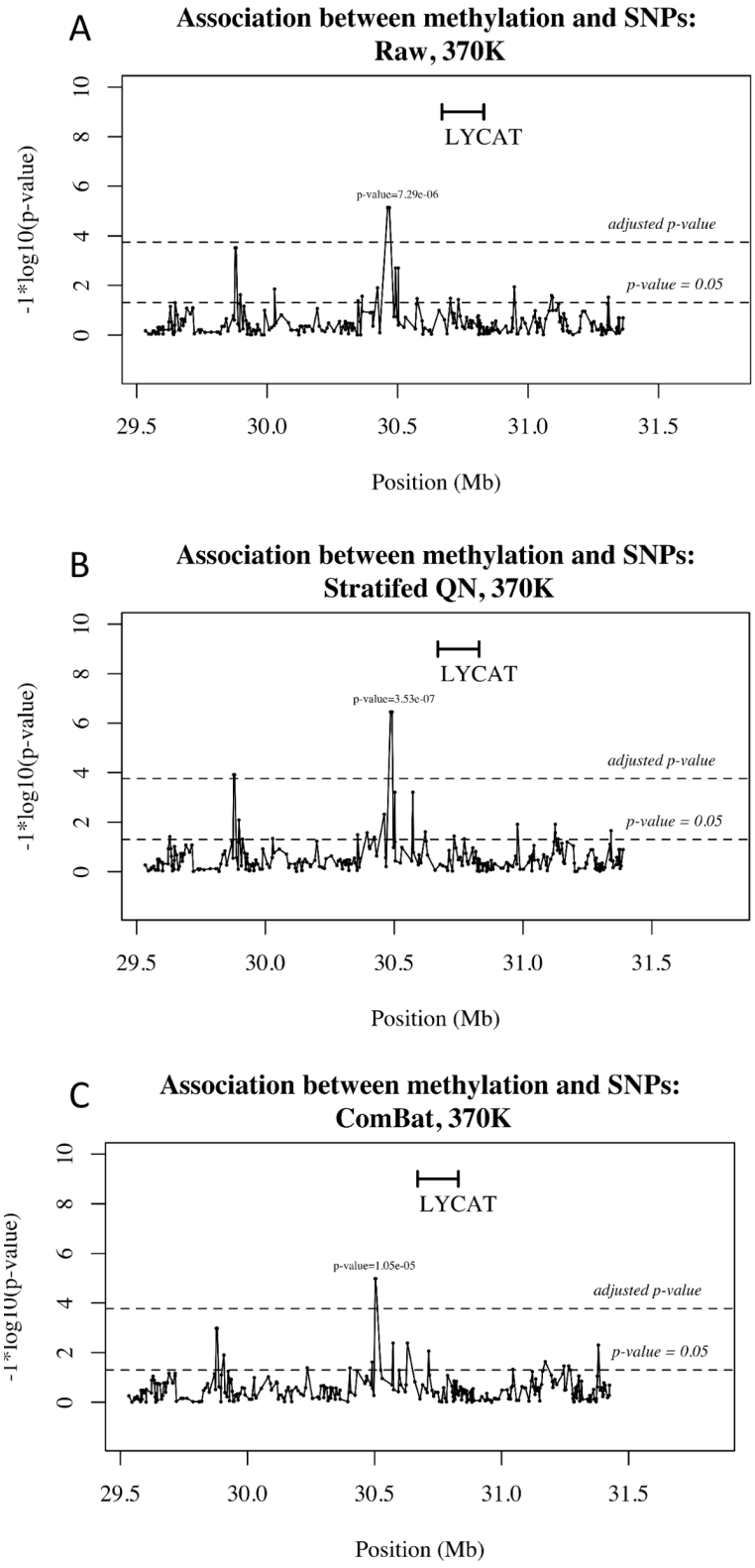


Figure 3.8 Association plot between SNPs and methylation.
Association between methylation at cg17749961 and SNPs in a 2Mb window promimal to the LYCAT gene. There is a significant association in the raw data (A, p-value=7.29e-06) which increases after stratified QN (B, p-value=3.53e-07) and drops slightly after ComBat correction (C, p-value=1.05e-05).

3.4 Discussion

There is currently a plethora of pre-processing methods and R packages available for analysis of methylation array data, and comprehensive review articles evaluating their utility have been published (Touleimat and Tost 2012; Marabita *et al.* 2013; Pidsley *et al.* 2013; Morris and Beck 2015). The majority of these are designed for specific types of sample sets, particularly those comprised of two distinct groups such as case-control or cancer-normal with substantial methylation differences between the two groups. For different data sets, such as those from familial studies, which include complex pedigree structures instead of two distinct groups, these methods may be ineffective or even worse, detrimental in that they introduce technical bias, as identified with selected methods in the analysis reported here. To correctly normalise data, it is critical to choose the most appropriate method, yet there has been little focus on developing appropriate processing pipelines for familial methylation array analysis, despite the current interest in inherited drivers of methylation patterns. Further barriers are the various format requirements and the lack of integration to provide a seamless processing pipeline. Here, eight different methods have been tested and a best-practice pre-processing pipeline presented for familial data (depicted in Figure 3.9). This pipeline creates a template to guide and expedite the analysis of familial datasets, particularly generated using the methylation array data.

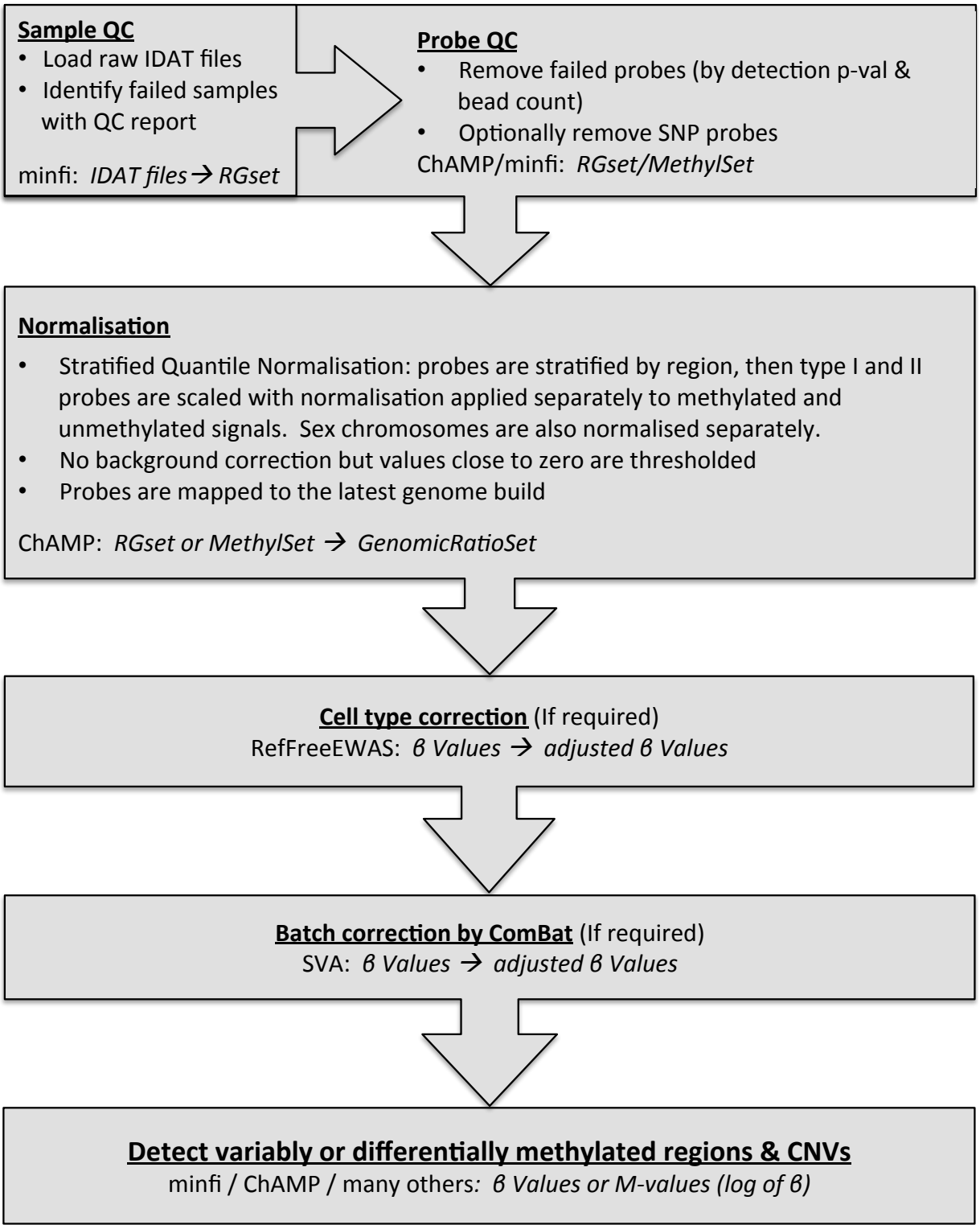


Figure 3.9 Pipeline for familial data processed on Illumina's 450K methylation array. Each box indicates a stage of the pipeline including the R package and the data format required/created in italics.

A fundamental requirement for processing methylation array data is effective adjustment for technical bias, including batch effects and adjusting for the two-probe biochemistry of the array. Batch effects may be introduced through bisulphite conversion or downstream processing or variation in array quality. Various methods have been developed to adjust for these effects, mostly involving variations in quantile normalisation, a technique commonly used in analysis of microarray datasets to align two different distributions so they result in identical statistical properties (Sun *et al.* 2011; Teschendorff *et al.* 2013; Aryee *et al.* 2014).

BMIQ and Functional Normalisation have been advocated as the preferred methods for analysis of cancer study data, as they are more specific in design than Quantile Normalisation and have been shown to be more effective at removing unwanted technical bias (Teschendorff *et al.* 2013; Fortin *et al.* 2014). However, while these methods were not specifically developed for such a purpose, they have been shown to work most effectively on case-control or tumour-normal datasets respectively. To the best available knowledge, optimal pre-processing methods for familial based data, such as performed here, have not been previously reported. Normalisation methods necessarily make assumptions about data, with the accuracy of these assumptions varying for different data sets. Thus the same normalisation method can have a vastly different effect on different types of data and conversely, as shown here, different normalisation methods can have vastly different effects on the same data. It is therefore key to select the right normalisation method for the data-set of interest. Of the eight methods tested, Stratified QN was consistently identified as the best normalisation method across all visual and quantitative evaluation metrics for

use in this context. The principle underpinning this normalisation is stratification by genomic region and is thus ideal for data where the differences between adjacent genomic loci are maintained. This is in contrast to tumour-normal tissue datasets where there are large blocks of dramatically altered methylation patterns throughout the tumour genome (Timp *et al.* 2014). Again not surprisingly, packages that utilize differences in negative control methylation patterns between cases and controls such as *FunNorm* were not found to be effective on familial datasets where no "normal" control is available.

The inherent strengths of familial data could be further exploited by a normalisation technique that accounts for known relationships between samples. Such a method could draw on pedigree information to ensure normalisation has effectively removed technical bias while maintaining known biologically relevant information such as relatedness and familial clustering by methylation. A diagnostic metric accounting for a known relationship could be used to test the efficacy of pre-processing methods in a similar manner to the standard error associated with imprinted differentially methylated regions (iDMRs) from the *wateRmelon* package.

It may also be of importance for researchers to consider the undesirable effect of non-specific binding and the presence of SNPs in the probe body. A study from the Weksberg lab found around 6% of probes on the array cross-hybridised to non-targeted genomic regions (Chen *et al.* 2013). They have catalogued these probes and suggest removing them prior to downstream analysis. Their study also demonstrates SNPs in the probe body can interfere with probe binding, altering the

methylation signal at around 14% of sites. Illumina recommends all probes containing a SNP within 10bp of the interrogated CpG site ought to be removed, while others suggest the 'probe effect' continues to the entire 50bp length of the probe (Zhi *et al.* 2013). The removal of all such probes would be undesirable for studies examining the effect of genotype on methylation, as evidence suggests the vast majority of these SNPs occur either at the CpG site itself (meSNPs) or close by (Shoemaker *et al.* 2010; Zhi *et al.* 2013).

To overcome this issue, Zhi and colleagues suggest an elegant approach to examine the effect of meSNPs on methylation without the potential bias introduced by SNPs altering probe binding (Zhi *et al.* 2013). The type II probes contain only one bead type for both methylated and unmethylated sites of interest, with the methylation status of the loci designated by the addition of a different coloured nucleotide (red or green) at the single base extension. As type II probes terminate one base pair before the cytosine of the CpG dinucleotide, a mutation at the cytosine itself would not affect probe binding. As such, probes without SNPs in the probe body but present at the single base extension can reliably be used to examine the effect of meSNPs on methylation, a very useful technique for examining the effect of inherited variation on methylation patterns.

Preservation of the biological integrity of information from methylation array data is imperative and requires appropriate pre-processing to minimize technical errors, which will be dictated by the type of data. Stratified QN in combination with ComBat batch correction performed the best of those methods tested for normalising

familial data interrogated on methylation array. This method was observed to remove technical biases while maintaining biologically relevant information; allowing true biological differences and similarities to inform the search for the role of methylation patterns driving disease processes. The workflow presented in Figure 3.9 outlines the methodology adopted to pre-process familial data in this study. This may also be instructive for other studies using familial data, including longitudinal studies where the same individuals are repeatedly measured over time.

Chapter.4 Identification and prioritisation of me-QTLs

4.1 Introduction

Although many factors influence DNA methylation patterns, including diet, environment and random factors; genetic variation has possibly the greatest effect on methylation, and this is observed when comparing different cell types, during development and across populations (Smith *et al.* 2014). Genetic variation is known to influence methylation patterns in a number of ways. Firstly, it has been observed for many years that coding variants in DNA methylating enzymes or chromatin remodelling factors affect epigenetic patterns across numerous diseases, particularly cancer (You and Jones 2012). Such coding variants in genes regulating the epigenetic environment alter protein function, impacting on the removal or addition of methylation marks across the genome. Secondly, variants in non-coding regions can affect DNA methylation patterns, although these regions have been less extensively studied, as their effect on gene regulation is more challenging to unravel (Barr and Misener 2016). As research efforts increasingly turn to understanding these regulatory elements, numerous studies are providing evidence for the contribution of non-coding genetic variants on DNA methylation patterns in various cancer types (Heyn 2016).

The mechanisms by which non-coding variants influence methylation patterns are complex. The variants termed *methylation quantitative trait loci* (meQTLs), may occur in three distinct locations, as indicated in Figure 4.1 below. *Cis*-meQTLs occur

in close proximity to the CpG site (within 1Megabase (Mb)) (Gamazon *et al.* 2012; Heyn *et al.* 2014; Rushton *et al.* 2015) while *trans*-meQTLs are located distally to the CpG site, influencing methylation over 1Mb away on the same chromosome or on different chromosomes, through 3-dimensional chromatin structures (Lemire *et al.* 2015). Finally, variants can also occur at the CpG sites themselves (CpG-SNPs) with numerous studies suggesting these may have the strongest effect on methylation patterns(Shoemaker *et al.* 2010; Zhi *et al.* 2013; Zhou *et al.* 2015).

Figure 4.1 Location of meQTLs.
SNPs affecting CpG methylation (meQTLs) have been described as occurring in *cis* within 1Mb of the CpG, at the CpG itself (CpG-SNP) or further than 1Mb from the CpG, in *trans.*

The location at which the meQTL occurs determines how it may affect methylation patterns. Most directly, CpG-SNPs can physically modify the cytosine of a CpG site to another nucleotide, most frequently thymine, removing the possibility of cytosine methylation (Hellman 2010; Gertz *et al.* 2011). Alternatively, variants may alter transcription factor binding sites, leading to inhibition of transcription factor binding and the recruitment of the transcription machinery (Kasowski *et al.* 2010), with altered transcription factor binding frequently shown to alter nearby methylation patterns (Kasowski *et al.* 2010; Banovich *et al.* 2014). The mechanism by which these altered methylation patterns occur is less well understood. However, it is now know that a transient decrease in expression is all that is required to initiate more permanent epigenetic silencing (Oyer *et al.* 2009). Lack of gene expression, including diminished occupancy at the DNA by the transcription machinery, leads to altered chromatin, namely a depletion of acetylated histones and methylated histone-3-lysine-4 (H3K4), and an enrichment of the methylated histone-3-lysine-9 (H3K9) histone mark (Yan *et al.* 2003). These altered histone modifications are then followed by increased DNA methylation, which facilitates a more permanently silenced gene expression state (Yan *et al.* 2003; Oyer *et al.* 2009).

Linked to this aberrant silencing, is the 'seeding hypothesis', which postulates that abnormal promoter methylation is seeded or extended from methylation in flanking regions such as shores and shelves (Graff *et al.* 1997; Deng *et al.* 1999; Hesson, Hitchins and Ward 2010). Such seeding of aberrant methylation has been described at the promoters of several tumour suppressor genes in various cancer types, including *MutL Homolog 1(MLH1)* in sporadic colorectal cancers and *Ras*

Associated Domain Family 1 Isoform A (*RASSF1A)* in sporadic breast cancer (Deng *et al.* 1999; Yan *et al.* 2003).

Another mechanism by which aberrant methylation patterns can be generated is through alterations in methyl-binding protein domains. This can affect the binding affinity of certain transcription factors such as the transcriptional repressor *CCCTC-Binding Factor (CTCF)* (Shukla *et al.* 2011), leading to the effects described above, or affect binding of factors like *methyl-CpG binding protein 2 (MeCP2)* and subsequent recruitment of chromatin remodelling complexes, which can lead to aberrant 3-dimensional chromatin structures. Aberrant binding of *MeCP2* can also affect alternative splicing, with a decrease in intragenic DNA methylation associated with altered splicing of transcripts (Maunakea *et al.* 2013).

Interestingly, CpG methylation is a significant driver of human polymorphisms, as methylated cytosines are highly vulnerable to deamination to thymine (Cooper and Youssoufian 1988; Rideout *et al.* 1990; Tomso and Bell 2003). As early as 1988, CpG-SNPs were described to play a key role in mutagenesis, accounting for 35% of coding mutations in one particular study, with 90% of these C→ T or G→A transitions (Cooper and Youssoufian 1988). Over a decade later, when non-coding variation was beginning to be examined, Tomso *et al.* surveyed two million non-coding SNPs for polymorphism "hotspots", and found that such variation was enriched outside CpG islands. They found that methylated CpGs outside islands, which are normally methylated, were 6.7 times more likely to contain SNPs than expected and that CpGs in islands (normally unmethylated) contained 6.8 fold less

variation than expected by chance. Finding comparable levels of C→T and G→A variants (80%), they concluded that methylated cytosine deamination plays a strong role in human variation and that variation within islands is supressed as these regions are normally un-methylated (Tomso and Bell 2003).

Genetic variation and CpG methylation are thus intrinsically linked, with the underlying genetic sequence establishing a 'propensity to methylate'; a predisposition or tendency towards a particular epigenetic pattern in the context of a certain genotype (Hesson, Hitchins and Ward 2010). These patterns are not solely dependent on genotype as they are also influenced by environmental and random factors, as described in Richard's 'facilitated epigenetic variation model' (Richards 2006). This genotype-epigenotype interaction is further described in Feinberg and Irizarry's 'inherited stochastic variation model' which proposes genetic sequence variation underlies the propensity for epigenetic variation, as certain DNA sequences are not only directly responsible for particular traits but also increase natural methylation variation for that trait (Feinberg and Irizarry 2010). Various stochastic and environmental factors then influence DNA methylation at these variably methylated regions (VMRs), resulting in increased phenotypic differences, which are then acted on by Darwinian selection in a similar manner to selection pressures affecting purely genetic traits. This model is supported by further studies examining distinct methylation patterns across different populations. For example, Heyn *et al.* found one-third of differential inter-individual methylation patterns to be independent of genotype, suggesting genetic and epigenetic evolutionary blueprints

are established and acted on by divergence and selection pressures resulting in phenotypic variation (Heyn *et al.* 2013).

A complete atlas of the influence of environmental and genomic variation on DNA methylation requires many more studies across various populations, tissue types and disease models. In the present study, genome-wide methylation and SNP profiles have been generated as described in Chapter 2. An appropriate pre-processing pipeline was developed to permit analysis of the genome-wide methylation profiles of familial prostate cancer cases and family members (Chapter 3, Cazaly et al, 2016). These pipelines have produced high quality methylation and genotype data for thirty-nine samples representing ten families. The output methylation profiles of these samples show significant familial grouping (Chapter 3 Figure.1F), indicative of the underlying inheritance of genetic drivers in these cohorts of families with high prostate cancer incidence.

Herein, these data will be interrogated to identify and prioritise prostate cancer relevant meQTLs. As the most influential variants are likely to be the CpG-SNPs themselves, this study focuses on these variants, adapting an elegant approach described by Zhi and colleagues who examined *cis*-meQTLs in the Genetics of Lipid Lowering Drugs and Diet Network data (Zhi *et al.* 2013). Specifically, Zhi et al. demonstrated that although the Infinium HumanMethylation 450k BeadChip was not designed to examine CpG-SNPs, careful selection of probes and analysis allows for accurate measurements at these sites. While SNPs located within probe bodies on the methylation array can interfere with binding of the probe to target DNA,

probes with a SNP at the single base extension (SBE) for Infinium type II probes can be reliably used to examine CpG-SNPs. Only Infinium type II probes are suitable for examining CpG-SNPs as the probes themselves do not cover the SBE. In contrast, Infinium type I probe bodies do cover the SBE and ought to be excluded. The authors report that more than 80% of genetic variants at CpGs (CpG-SNPs) are meQTLs and that the influence of these variants extends beyond the CpG-SNP, concluding these are important determinants genotype-linked epigenetic changes.

4.2 Methods

Two distinct approaches have been employed to determine prostate cancer relevant meQTLs, adopting several key strategies. The first approach prioritises CpG sites with highly variable methylation between individuals, which is predicted to vary with genotype. It is hypothesised that, as the samples are drawn from families with elevated rates of prostate cancer incidence, the variants driving predisposition will be enriched in the affected men, influencing aberrant methylation patterns and will be distinguishable from their unaffected relatives. The second approach draws on previously identified regions of the genome associated with prostate cancer risk, with the hypothesis that a proportion of the unexplained risk variants previously identified in both familial and sporadic prostate cancer studies, may be due to non-coding variants driving aberrant methylation profiles. The pipeline for identifying variants of interest from each approach, together with subsequent meQTL testing and prioritisation can be seen in Figure 4.2.

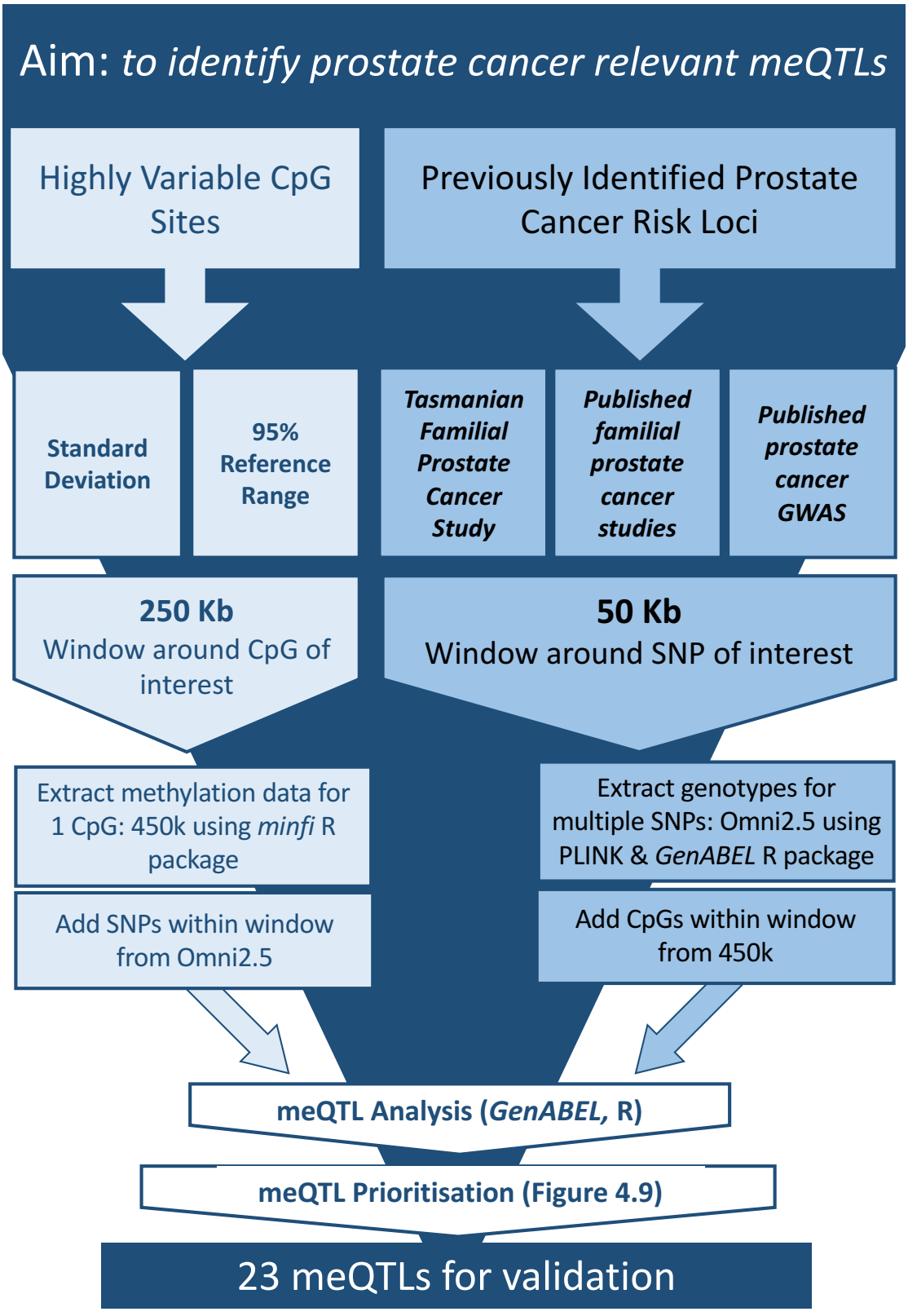


Figure 4.2 The pipeline employed in this study to identify prostate cancer relevant meQTLs. Two distinct approaches were taken to identify meQTLs. Regions of interest from both approaches were analysed in the same manner with the *GenABEL* R package and prioritised using the same metrics as described in Figure 4.9.

4.2.1. Identification of CpG sites with highly variable methylation

Allele-specific methylation profiles consist of three distinct methylation levels, namely clusters of high, mid and low methylation levels, corresponding to homozygous genotypes driving high and low methylation and heterozygous genotypes driving intermediate methylation (Deng *et al.* 2009). A heterozygous genotype has approximately 50% methylation as one allele is methylated and the other un-methylated across a population of cells. As such, variable CpG sites in this study were selected by choosing those displaying three distinct clusters of methylation between samples, with the aim of examining genotype driven meQTLs.

Given that it has been reported that CpG-SNPs are likely important drivers of methylation changes, 7,049 probes on the methylation array interrogating a known CpG-SNP were selected and then prioritised on methylation variability at the CpG-SNP. This method ranked CpG sites by standard deviation from the mean methylation value across individuals at each site (base *stats* R package, R Core Team 2016). This identification was enabled through the *annotation* function in the *minfi* R package (Aryee *et al.* 2014), which annotates genomic information, including CpG-SNPs, to the methylation data. Specifically, the standard deviation at each CpG-SNP was calculated between individuals, and the 100 CpG-SNPs with the greatest standard deviation were selected for meQTL analysis. Appendix 4.5 contains details of the R code used in this analysis.

Secondly, to more widely examine variable methylation outside CpG-SNPs, an alternative method was used on all probes. In this approach the central distribution

of methylated values was captured for each CpG site using the *quantile* function from the *stats* R package by selecting the 95% Reference Range. This method has previously been described by Lemire and colleagues to examine *trans*-meQTLs in lymphocytes (Lemire *et al.* 2015). Specifically, the difference between the 97.5% quintile and the 2.5% quintile is calculated at a given CpG site. Adopting this approach is equivalent to calculating the range of methylation values, after the upper and lower 2.5% of values are discarded. This is to limit the effect of outliers, which may skew results if included.

4.2.2. Selection of prostate cancer risk loci for meQTL analysis

In this approach, the association between loci within previously identified prostate cancer risk regions and neighbouring CpG methylation was examined. Previously examined prostate cancer risk regions were selected from three different sources; 1) regions previously identified through modified linkage analysis in this laboratory using the Tasmanian Familial Prostate Cancer Resource, 2) other published familial prostate cancer risk loci from the OMIM resource, and 3) risk SNPs identified by GWAS published on NCBI's GWAS Catalogue.

4.2.2.1 Risk loci identified through the Tasmanian Familial Prostate Cancer Study

A previous modified linkage analysis in this laboratory using BEAGLE (Browning and Browning 2011) on the Tasmanian Familial Prostate Cancer Study, identified four regions generating modified logarithm of the odds (LOD) scores . These regions were of nominal genome-wide significance (unpublished). Figure 4.3 illustrates the

Manhattan plot for this study, showing the significance of prostate cancer risk loci genome-wide.

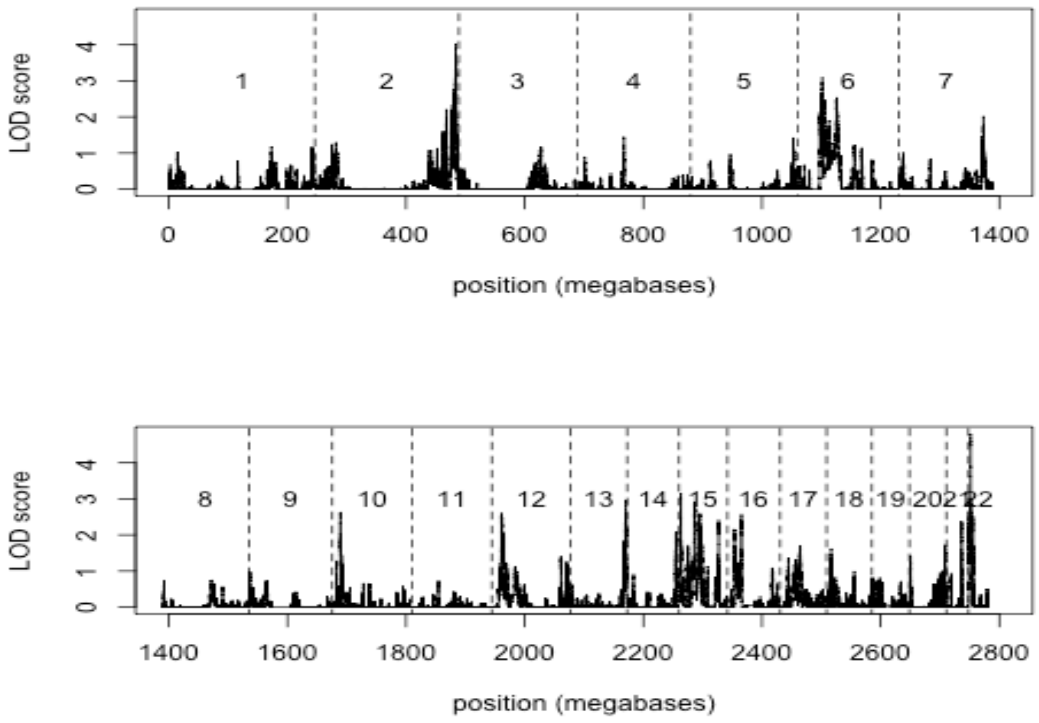


Figure 4.3 Manhattan Plot highlighting prostate cancer linkage regions.
A SNP genome wide scan using the Affymetrix CNV370 array using *Tasmanian Familial Prostate Cancer* dataset comprising 265 individuals of which 171 were affected cases, and one or two offspring of deceased cases representing a further 71 cases. Plot generated using IBD sharing calculated using the fast IBD sharing option in BEAGLE for 265 individuals representing prostate cancer cases and children of diseases cases. A reference group of 373 unrelated Tasmanian controls was used as a control dataset. Each individual chromosome is numbered. Note, the major histocompatibility complex (MHC) region has been removed in this graph due to complexity and linkage disequilibrium in this region causing false signal.

These nominal linkage regions, also listed in Appendix 4.1 contain 143 SNPs on four chromosomes; fifty on chromosome 2, forty-nine on chromosome 6, twenty-nine on chromosome 15 and fifteen on chromosome 22. Modified LOD scores ranged from 1.98 to 4.8, with the majority of scores over 3, indicating an increased probability of genetic linkage to disease. As these loci were annotated to a previous genome build, *SNPnexus* (http://www.snp-nexus.org) was used to convert the genomic positions to the hg19 build co-ordinates so comparison to methylation array data and other prostate cancer risk data could be conducted.

4.2.2.2 Risk Loci identified through published familial prostate cancer studies

Thirty-two prostate cancer risk regions were selected from the *Online Mendelian Inheritance in Man* (OMIM) database (http://www.ncbi.nlm.nih.gov/omim). OMIM is a freely available compendium of human genes and phenotypes. This database is regularly updated and curated by the McKusick-Nathans Institute of Genetic Medicine at the Johns Hopkins University. As the regions were already annotated to the hg19 genome build no genome conversion was required. Unlike the risk loci from the two other databases, which could be used to generate a 50Kb window for analysis, this dataset contained large genomic regions requiring a sliding window approach to allow all loci in the regions to be sequentially tested without inflating the multiple testing burden within each window. See Figure 4.4 below for a schematic of the sliding window design.

4.2.2.3 Risk Loci identified through published prostate cancer GWAS

A list of 320 prostate cancer risk SNPs from published GWAS was generated using the GWAS catalogue (https://www.ebi.ac.uk/gwas/), a joint venture founded in 2008 by the National Human Genome Research Institute and the European Bioinformatics Institute. The resource is a manually curated and quality controlled collection of published GWAS with over 100,000 SNPs and associated traits. Sixty-three duplicate SNPs from multiple studies were removed, along with 1 SNP with no annotation information on UCSC, leaving 256 unique SNPs. As these SNPs were annotated to the latest genome build, hg38, they were then passed through *SNPnexus* to generate hg19 genomic co-ordinates.

4.2.3 Filtering of CpG probes prior to meQTL association

As discussed in section 4.1, SNPs located in the probe body on the methylation array can interfere with binding of the probe to target DNA, creating an artificial methylation signal and technical bias in the data (Chen *et al.* 2013). As such, these probes were removed from further analysis.

4.2.4 Association between SNPs and CpGs in identified risk windows

For each method of determining variable CpG sites (standard deviation and 95% reference range), the 100 most variable CpGs were selected for meQTL analysis. Genomic windows of 250Kb were generated around each CpG, with SNPs located within each genomic window extracted from the genotype array using PLINK (Purcell *et al.* 2007) and the *GenABEL* R package (GenABEL project developers 2013). These SNPs were then analysed against one variable CpG per window.

For previously identified prostate cancer risk loci identified from the Tasmanian Familial Prostate Cancer Study and published GWAS, a 50Kb window was drawn around each SNP of interest with all CpG sites falling within that genomic range drawn from the methylation array using the *minfi* R package (Aryee *et al.* 2014). These were analysed against all SNPs from the genotyping array in the same window (instead of only the risk SNP), to allow for the possibility that the previously identified SNP was not the causative risk SNP, but was in linkage disequilibrium with the nearby risk SNP**.** The approach for the risk loci identified through published familial prostate cancer studies was slightly different as this data covered large genomic windows, rather than single nucleotide variants. For this approach a sliding window, as described in Figure 4.4 below was used, with every SNP and CpG within each window extracted from the methylation and genotype arrays as discussed above.

The window sizes were carefully chosen to include as many informative SNPs and CpGs as possible without being unduly large to limit discovery power by excessive statistical tests. Window sizes were selected in accordance with previous studies, which have found meQTL associations typically extend 10-15Kb (Zhi *et al.* 2013; Smith *et al.* 2014). Luijk and colleagues also suggest windows between 10-50Kb, as larger windows of several hundred base pairs can lead to prohibitive false discovery rates as the number of statistical tests are extremely high (Luijk *et al.* 2015).

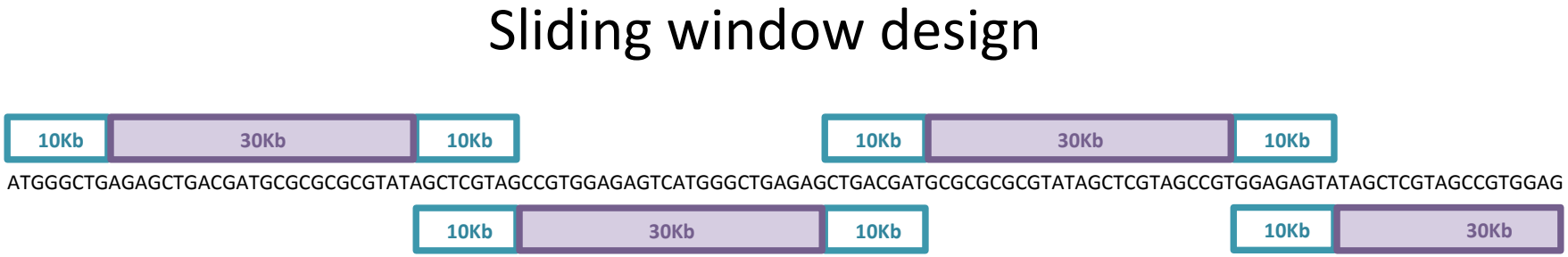


Figure 4.4 Sliding window design.
A sliding window strategy was used to analyse prostate cancer associated genetic regions identified through the familial linkage approach to generate genomic regions for association analysis. A unique region of 30Kb with a 10Kb overlap either side was chosen to create a 50Kb window.

A polygenic model using a hierarchical generalized linear model accounting for kinship was used from the *GenABEL* package to examine the association between each SNP and each CpG in each window. Kinship coefficients for the thirty-nine samples (listed in Table 2.1) were calculated with the *Identity-by-State* function in the *GenABEL* R package using a direct IBS computation. Negative $\log_{10}$ transformations of the p-values were taken to create more interpretable values, as the differences between p-values are more apparent. Using a negative $\log_{10}$ transformation, a significant p-value of 0.05 translates to 1.3. However, as there are many statistical tests required in genomic studies, convention is to correct for multiple testing error across the genome adjusting the significance threshold to <5x10^(-8) which equates to a negative log10 p-value of 7.3 (Panagiotou, Ioannidis and Project 2012).

In the current study, the significance threshold was adjusted by Bonferroni Correction, with a typical significant p-value of 0.05 divided by the number of statistical tests (ie. SNP x CpG combinations) in each window. These adjusted thresholds are displayed on the association plots in this chapter. However, a more stringent cut-off was employed when considering meQTLs for the prioritisation pipeline. To qualify for the pipeline, only meQTLs with transformed negative $\log_{10}$ p-value above 10 (which is much higher than the conventionally used genome-wide significance adjustment) were considered. Given that the total number of tests carried out in this study is smaller than the number of tests in a genome wide association study, all meQTLS that were considered for the prioritisation pipeline were significant after adjusting for multiple testing. Regions were then plotted in a

similar manner to GWAS Manhattan plots using the generic plot function in R. Appendices 4.5 and 4.6 detail the R code used to perform the identification and prioritisation of meQTLs.

4.2.5 meQTL Prioritisation

MeQTLs were considered for the prioritisation pipeline if the association between SNPs and methylation was above the adjusted significance threshold of 10 for negative $\log_{10}$ p-values. Four filtering steps were then applied to the significant associations as detailed below.

4.3 Results

Appropriate quality methylation and genotype data obtained from thirty-nine samples was analysed to identify and prioritise prostate cancer relevant meQTLs. Samples included in these analyses are indicated by orange stars in Figure 4.5 below, and listed in Table 2.1 of Chapter 2 (sex and disease status information is included).

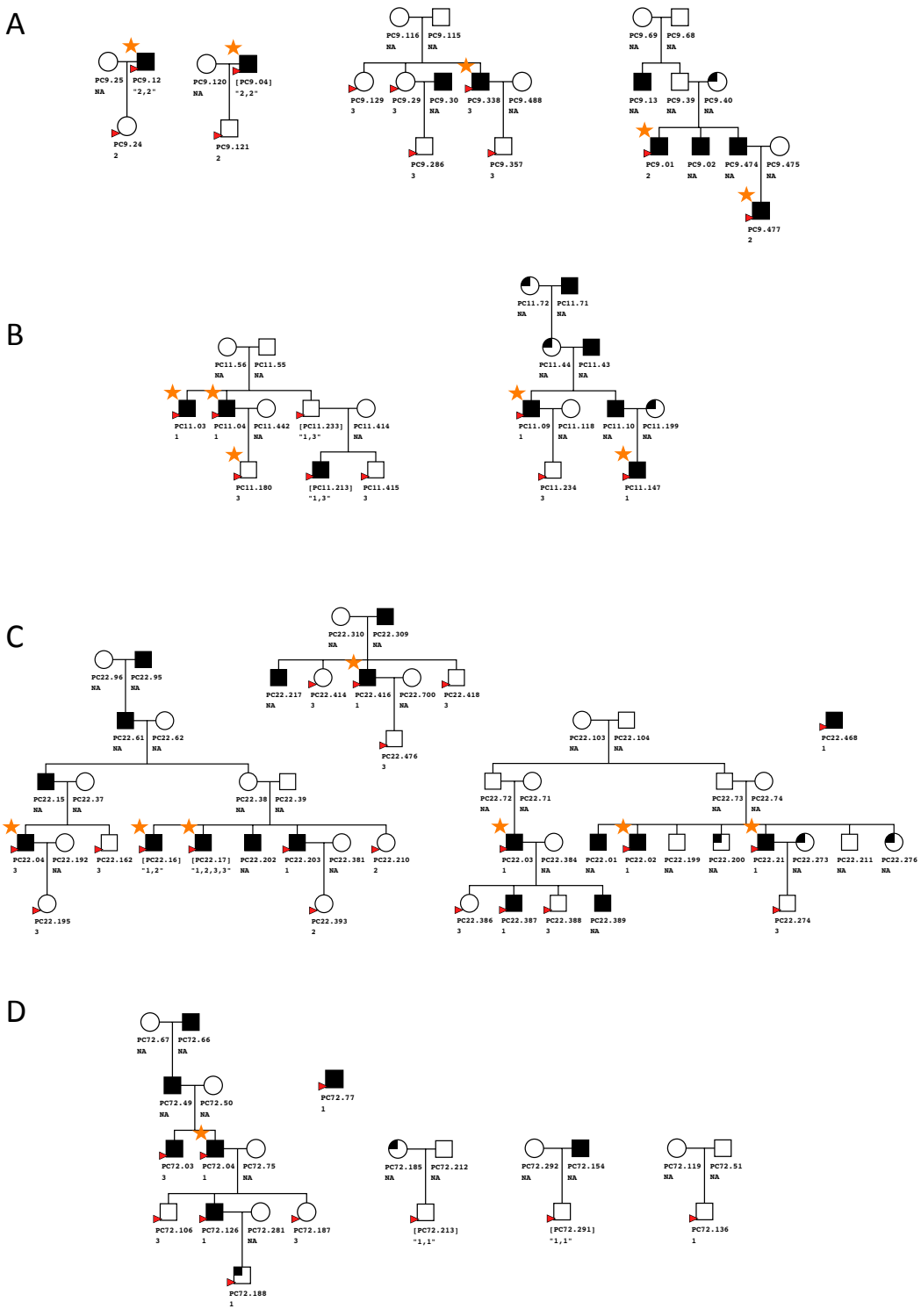


Figure 4.5 Pedigree Information for samples included in meQTL Analysis .
Four clusters from Family 9 (A), two from Family 11 (B), four from Family 22 (C), and five from Family 72 (D) were selected for meQTL analysis**.** Circles represent women and squares men, with individuals affected by prostate cancer filled in black, those unaffected unfilled with individuals affected by other cancers quarter filled. Samples analysed are indicated by an arrow-head with thirty-nine samples with both good quality methylation and Genotype data indicated by orange stars.

4.3.1 Identification of CpG sites with highly variable methylation

To determine the most variable methylation sites between individuals, two approaches were taken. Firstly, the standard deviation of methylation values at probes containing CpG-SNPs was examined, with the 100 most variable CpGs tested for genotype-methylation associations. Significant meQTL associations (threshold adjusted by Bonferroni Correction) were detected in each of the 100 windows analysed, with 93 windows containing a $-\log_{10}$(p-value) above the stringent adjusted significance threshold of 10. The most variable 100 CpG sites together with their $-\log_{10}$(p-values) and the number of significant associations per window are presented in Appendix 4.2, with summary statistics provided in Table 4.1.

Table 4.1 Summary of prioritisation and association analysis for the variable methylation approach

Variable Methylation Approach	Standard Deviation	95%-Reference Range
Number of windows from each approach	100	100
Windows for which an meQTL association was generated	100	98
Windows with $-\log_{10}$(p-value) above the adjusted significance threshold using Bonferroni Correction	100	96
Mean number of significant associations per window using Bonferroni Correction	22	23
Number of CpG-SNPs with a $-\log_{10}$(p-value) above the adjusted significance using Bonferroni Correction	100	90
Highest $-\log_{10}$(p-value)	33.8	30.25
Mean $-\log_{10}$(p-value)	22	19
Windows with a $-\log_{10}$(p-value) above the adjusted significance threshold of 10	93	85
Number of CpG-SNPs with a $-\log_{10}$(p-value) above the adjusted significance threshold of 10	93	81
Overlap with alternate variable methylation approach	67	67
Overlap with risk loci approach	3	3

A second approach was then taken to examine the influence of meQTLs outside of CpG-SNPs and to provide validation of CpGs identified in the first method. The 95% Reference Range was calculated for all probes as described in section 4.3.1, with ninety-eight genomic windows successfully generated. The vast majority of these (ninety-six) produced $-\log_{10}$(p-values) above the Bonferroni adjusted significance threshold, with eighty-five above the stringent adjusted significance threshold of 10. The ninety-six CpGs with $-\log_{10}$(p-values) above the Bonferroni adjusted significance threshold, together with the $-\log_{10}$(p-values) and the number of significant associations per window are detailed in Appendix 4.3, while summary statistics presented in Table 4.1.

Sixty-seven of the identified meQTLs were identified by both methods, (highlighted in orange in Appendix 4.2). Both methods also had analogous maximum $-\log_{10}$(p-values) (33.80 for Standard Deviation and 30.25 for 95%-Reference) and mean $-\log_{10}$(p-values) (22 for Standard Deviation and 19 for 95%-Reference) as well as a similar number of significant associations above the adjusted Bonferroni Correction per window (mean of 22 for Standard Deviation and 23 for 95%-Reference Range). There were thirty-three CpGs that did not overlap between the two methods, including the most significant cg13387643 which had a $-\log_{10}$(p-value) of 33.80. This CpG was identified as highly variable by the standard deviation approach, yet was only included in the top 200 most variable sites determined by the 95% Reference Range. As such, meQTL analysis would not have been performed on this CpG if it had not been detected in the standard deviation approach. Additionally, only ten of the

top twenty most variable CpGs identified by the standard deviation method were ranked in the top 100 CpGs identified by the 95% Reference Range method.

4.3.2 Selection of prostate cancer risk loci for meQTL analysis

To identify genomic regions pertinent to prostate cancer predisposition, three data sources of previously identified prostate cancer risk loci were utilised to create 6108 genomic windows.

A previous linkage analysis in our laboratory using the Tasmanian Familial Prostate Cancer Resource, identified four regions with nominally significant modified LOD scores associated with prostate cancer risk and a summary of the statistically significant associations is presented in Table 4.2, Risk Loci[1]. Additionally, thirty-two prostate cancer risk regions previously associated with familial prostate cancer were examined and these are presented in Table 4.2, Risk Loci[2]**.** There is evidence that prostate cancer susceptibility variants identified by GWAS also contribute to hereditary prostate cancer (Teerlink *et al.* 2014). As such, previously identified prostate cancer risk SNPs from GWAS were also examined in this approach (Table 4.2, Risk Loci[3]). Pooling the putative prostate cancer susceptibility loci identified from the three sources generated 6108 windows and, association analysis between methylation and genotype was successfully performed in 4994 windows. Three-hundred and seventy-two windows contained meQTL $-\log_{10}$(p-value) above the adjusted significance threshold. The fifty most significant associations identified from the three prostate cancer risk loci methods is presented in Appendix 4.4.

Table 4.2 Summary of prioritisation and association analysis for the prostate cancer risk loci approach

Prostate Cancer Risk Loci Approach	*Risk Loci* [1]	*Risk Loci* [2]	*Risk Loci* [3]	*Total*
Number of windows	143	5723	242	6108
Windows for which an meQTL association was generated	143	4613	238	4994
Windows with a $-\log_{10}$(p-value) above the adjusted significance using Bonferroni Correction	13 (9%)	228 (5%)	131 (55%)	372
Windows with a $-\log_{10}$(p-value) above the adjusted significance threshold of 10	*4*	*8*	*36*	48
Number of CpG-SNPs with a $-\log_{10}$(p-value) above the adjusted significance using Bonferroni Correction	6	16	30	52
Number of CpG- with a $-\log_{10}$(p-value) above the adjusted significance threshold of 10	4	7	22	33 (69%)
Overlap with variable methylation approach	0	3	0	3

[1] *Risk loci identified through the Tasmanian Familial Prostate Cancer Study*
[2] *Risk Loci identified through published familial prostate cancer studies- taken from OMIM database*
[3] *Risk Loci identified through published prostate cancer GWAS- taken from GWAS Catalogue*

4.3.3 Association between genotype and methylation

Following generation of genomic windows encompassing the selected CpGs or SNPs (as detailed previously), association analyses were performed in each window using a polygenic linear model to identify meQTLs. MeQTLs with a $-\log_{10}$(p-value) above the adjusted significance threshold of 10 were subsequently prioritised through several filtering steps as described below.

To determine if meQTL associations were associations detected by chance or possibly an artefact of the methodology employed, ten genomic regions were chosen at random across the genome, and tested for association between genotype and methylation. The 10 plots are presented in Figure 4.6 (A-J) together with the number of associations above the adjusted significance threshold. Of these regions, one showed a statistically significant association (Figure 4.6 F) while the remainder contained no significant associations, providing evidence that the significant associations generated by the analyses outlined here are true biological associations and not the result of methodological bias. Additionally, the region that returned a significant association was found to contain a SNP in the probe body, which may have affected the probe binding, inflating the methylation signal. For the ongoing analyses, all such probes with potentially spurious binding were removed during the prioritisation pipeline.

In contrast, many significant associations were generated from both variable methylation approaches and the risk loci approaches. A visual representation of a significant association analysis is shown in Figure 4.7. As indicative on the smaller

inset plot, methylation levels at the variable CpG of interest are clustered by genotype. Methylation at three neighbouring CpGs either side of the variable CpG was examined, yet no altered methylation pattern was observed (see Figure 4.8).

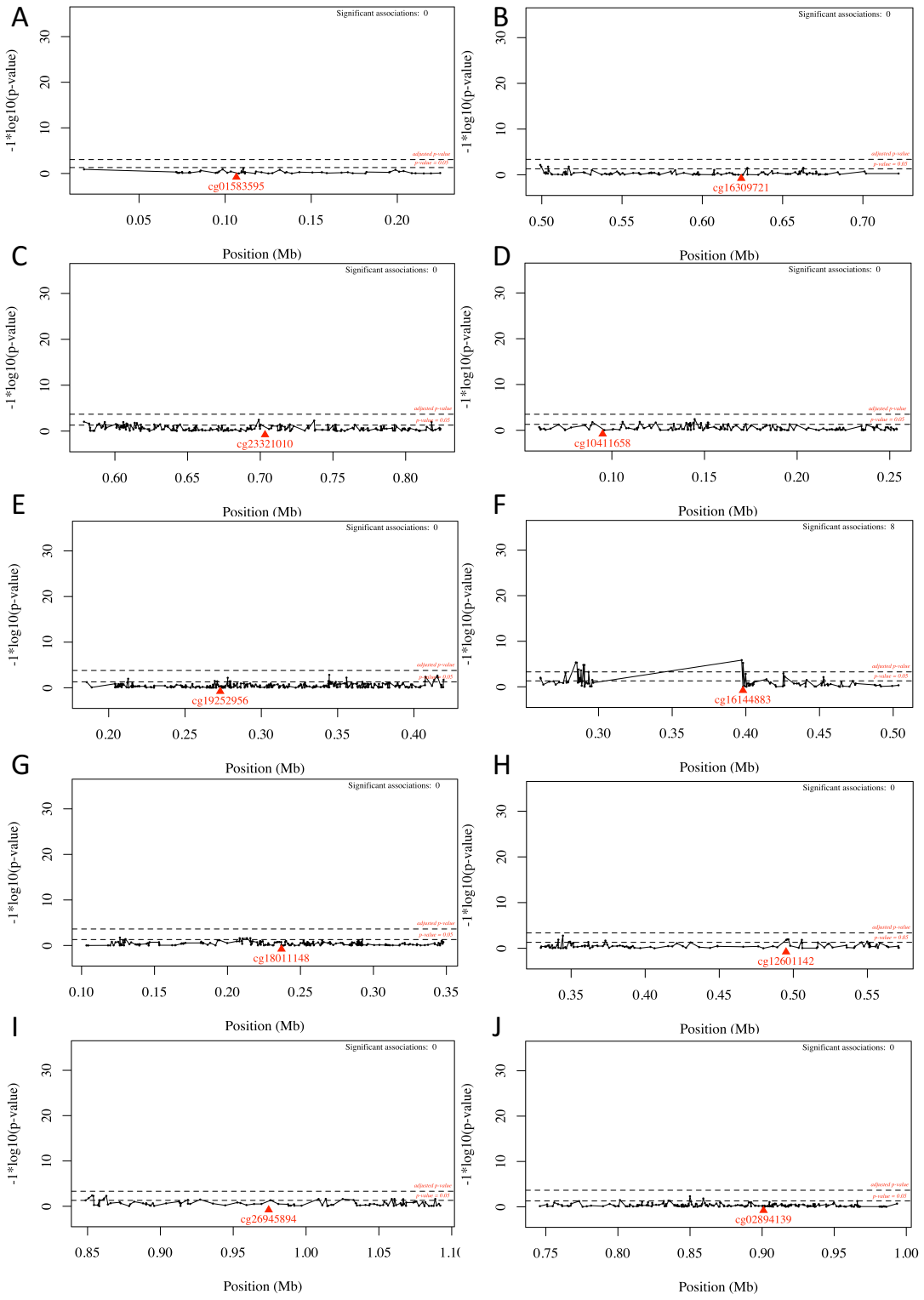


Figure 4.6 Association between Methylation and Genotype: Negative control plots. Ten genomic regions of 250Kb were chosen at random and association analysis was performed between SNPs and CpGs in the regions. Genomic location in Mb is presented on each x-axis with the $-\log_{10}$(p-value) on the y-axis. The location of each CpG is indicated by a red triangle with 0.05 p-value ($-\log_{10}$ of 1.3) and Bonferroni adjusted p-value thresholds indicated with dashed horizontal lines. Only F shows a significant genotype-methylation association.

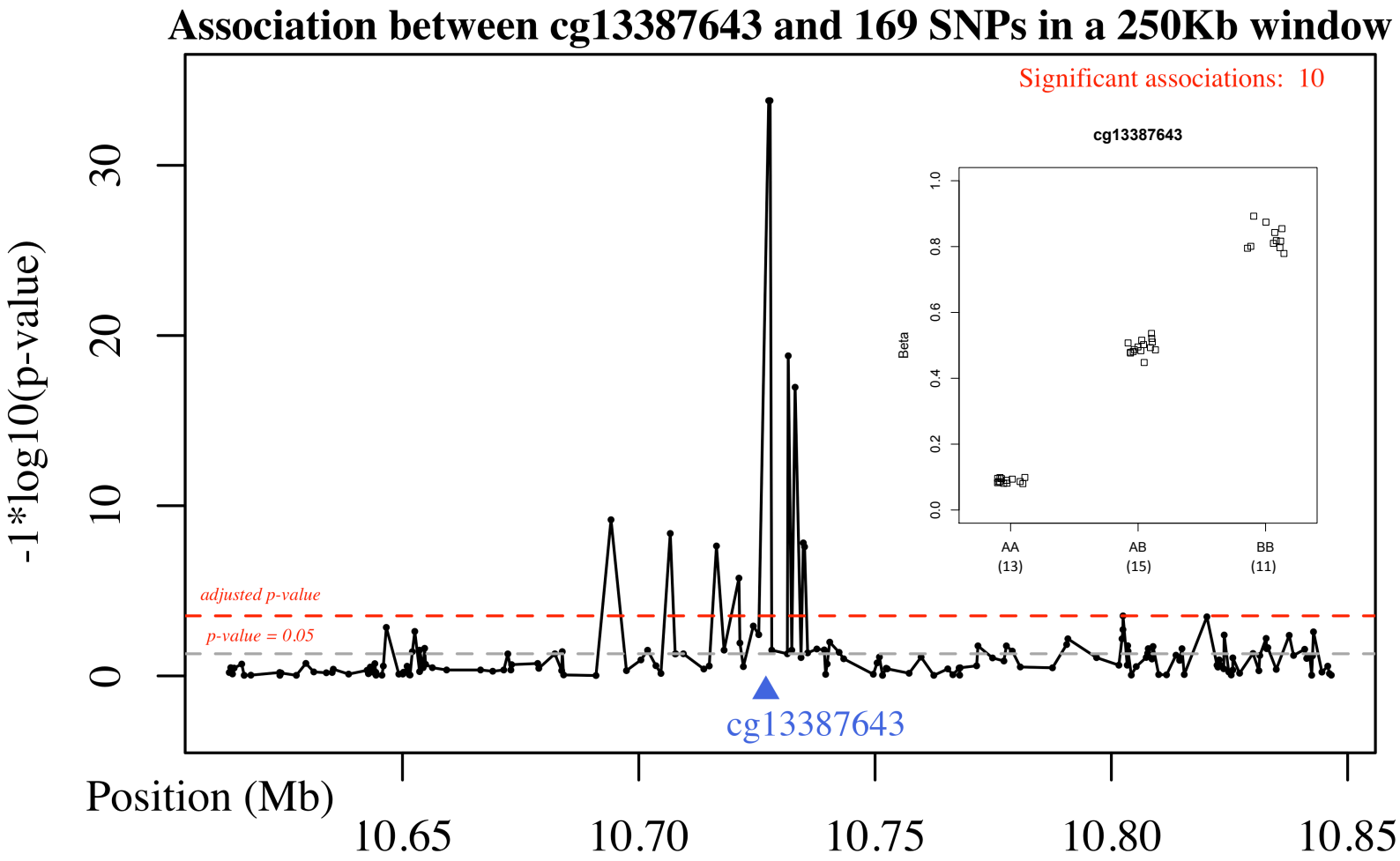


Figure 4.7 Visualisation of association between methylation and genotype: Significant association at cg13387643.
In this 250Kb region, genotypes at 169 SNPs were compared to methylation levels at one variable CpG located in the centre of the region, indicated by the blue triangle. 10 SNPs had $-\log_{10}$(p-values) above the Bonferroni adjusted significance p-value threshold. The smaller inset plot shows methylation level at this CpG with samples grouped by genotype at the closest SNP.

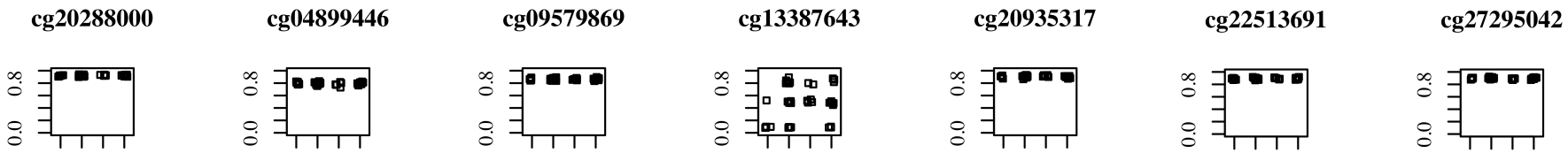


Figure 4.8 Methylation surrounding the most significant CpG-SNP. Methylation levels at three neighbouring CpGs either side of the most significant CpG-SNP (cg13387643). Samples in each of the surrounding CpG sites have high levels of methylation tightly clustered together.

4.3.4 meQTL Filtering and Prioritisation

To prioritise the most disease-relevant variants, four levels of filtering were applied to significantly associated loci. Figure 4.9 highlights the four filtering stages, together with the number of CpGs progressing through each level. In the first instance, meQTLs with a $-\log_{10}$(p-value) below the adjusted significance threshold of 10 were removed from further analysis, leaving the 111 most significant associations, with forty-eight unique CpGs from the risk loci approach and sixty-six from the variable methylation approach, with three CpGs overlapping between the variable methylation and risk loci approaches.

To maintain the focus on CpG-SNPs, loci lacking a CpG-SNP were removed. Ninety-nine CpGs remained, sixty-six from the variable methylation approach and thirty-three from the risk loci approach. For each of these ninety-nine CpGs the function of the nearest gene was examined. While CpG methylation and other regulatory motifs do not always affect expression at the most proximal gene, instead influencing expression at distal regions (in *trans*), current analysis tools make it extremely challenging to accurately examine regulatory function outside proximal genes. Genes with a reported role as a tumour suppressor or oncogene, a role in proliferation, cell cycle regulation, angiogenesis or gene regulation were included. Illumina's genomic annotation on the methylation array was used to determine the most proximal gene, namely if CpGs were located within a gene or within 1,500bp of a transcription start site. These annotations were then confirmed using the UCSC genome browser. CpG sites that were not annotated to a gene on the methylation array were excluded from further analysis. Thirty-seven CpG sites remained after this filtering, and are

presented in Table 4.3 together with a subset of the annotation information and reported functionality of the nearest gene.

Table 4.3 The 37 most significant associations from all approaches.
CpGs are filtered by significance of p-value, presence of CpG-SNP, absence of probe SNP and gene function. CpG sites prioritised for validation are highlighted in orange with three CpG sites prioritised from both approaches highlighted in yellow.

A) meQTLs identified throughstandard deviation

	CpG Name	In other variable approach	CpG-SNP	Gene	Gene Function
1	cg13387643	**no**	rs284310	CASZ1	Zinc finger transcription factor, may function as tumor suppressor
2	cg09084244	**no**	rs1109559	CDK2AP1	Possible regulatory role in DNA replication. Forms core subunit of NURD complex; epigenetically regulates embryonic stem cell differentiation. Associated with oral cancer
3	cg25013753	**no**	rs1051508	ARHGAP22	Regulates cell motility & angiogenesis. May be involved in transcription regulation via interaction with VEZF1
4	cg08210706	**no**	rs10135403	SERPINA5	Inhibits urinary-type plasminogen activator-dependent tumor cell invasion and metastasis
5	cg00231519	**no**	rs36101953	C10orf46	Cell cycle associated protein capable of promoting cell proliferation
6	cg02978201	**no**	rs737008	PRM1	Protamines substitute for histones in the chromatin of sperm during the haploid phase of spermatogenesis
7	cg00345083	**no**	rs7517857	AJAP1	Plays a role in cell adhesion and cell migration
8	cg08238375	**no**	rs4705795	MCC	Candidate colorectal tumor suppressor gene thought to negatively regulate cell cycle progression

B) meQTLs identified through 95% Reference Range

	CpG Name	In other variable approach	CpG-SNP	Gene	Gene Function
9	cg20592836	yes	rs2378256	TP53INP2	Dual role as transcription factor & in autophagy
10	cg02464073	no	rs1721	ITGB2	Important role in immune response. Defects in gene cause leukocyte adhesion deficiency & gastrointestinal carcinoma
11	cg08146865	yes	rs3197223	NME6	Inhibitor of p53-induced apoptosis
12	cg23698271	no	rs11199030	TIAL1	RNA-binding protein, regulates various activities including translational control, splicing & apoptosis
13	cg21927991	yes	rs5025124	ZFAT	Puatively binds DNA & functions as a transcriptional regulator involved in apoptosis and cell survival
14	cg07240846	yes	rs10906142	CAMK1	Activates transcription factor CREB1 in hippocampal neuron nuclei, possible involvement in apoptosis of erythroleukemia cells
15	cg06330797	yes	rs7357046	RPS6KA2	Implicated in cell growth & differentiation, may function as tumor suppressor in ovarian cancer
16	cg19360212	yes	rs11200296	NSMCE4A	Involved in DNA double-strand breaks by homologous recombination, required for telomere maintenance, involved in positive regulation of response to DNA damage stimulus
17	cg05331763	no	rs79974293	FOXK2	Related pathways: Cell Cycle / Checkpoint Control and Wnt / Hedgehog / Notch. GO annotations include sequence-specific DNA binding transcription factor activity & magnesium ion binding
18	cg01891583	yes	rs2304466	USP7	May induce p53/TP53-dependent cell growth repression & apoptosis
19	cg05161773	yes	rs426439	S EPT9	Involved in cytokinesis & cell cycle control, possible ovarian tumor suppressor gene. Chromosomal translocation results in

					acute myelomonocytic leukemia
20	cg11251367	yes	rs12403072	FMN2	Role in organization of cytoskeleton & cell polarity. Involved in responses to DNA damage
21	cg09993319	yes	rs7898151	MGMT	Involved in Cell Cycle & DNA-methyltransferase activity
22	cg04610028	yes	rs2967607	RAB11B	Involved in regulating exocytotic and endocytotic pathways
23	cg05338731	no	rs5751591	RAB36	Associated with rhabdoid cancer
24	cg02658043	yes	rs7465214	NRBP2	Associated with medulloblastoma
25	cg26365090	yes	rs11700304	TOX2	RNA polymerase II transcription factor binding. Putative transcriptional activator
26	cg17662493	yes	rs6006744	SMC1B	Involved in chromatid cohesion and DNA recombination during meiosis and mitosis
27	cg03796003	yes	rs117229426	KCTD5	Associated with rectal neoplasm
28	cg09289202	yes	rs6757649	STK25	Possible role in response to environmental stress and cell migration

C) meQTLs identified through the Tasmanian Familial Prostate Cancer Study

	CpG Name	Variable Approach	CpG-SNP	Gene	Gene Function
29	cg03036702	No	rs4714482	FOXP4	Transcription factor, may play a role in tumors of the kidney & larynx
30	cg16995742	No	rs7577630	COPS8	Important regulator in multiple signaling pathways

D) meQTLs identified through published familial prostate cancer studies

	CpG Name	Variable Approach	CpG-SNP	Gene	Gene Function
	cg11251367	Yes	rs12403072	FMN2	Role in organization of actin cytoskeleton & cell polarity. Involved in responses to DNA damage, cellular stress & hypoxia
31	cg00069771	No	rs17452776	C1orf57	Involved in purine metabolism
32	cg23209941	No	rs12137417	DISC1-TSNAX	DISC1: involved in neurite outgrowth & cortical development
33	cg07134368	No	rs12034296	TSNAX-DISC1	TSNAX: Interacts with DNA-binding protein that binds consensus sequences at breakpoint junctions of chromosomal translocations

E) meQTLs identified through published prostate cancer GWAS

	CpG Name	Variable Approach	CpG-SNP	Gene	Gene Function
34	cg23069046	No	rs6920276	REXO2	Possible role in DNA repair, replication, recombination
	cg03036702	No	rs4714482	FOXP4	Transcription factor, implicated in kidney & larynx tumors
35	cg09349613	No	rs72828989	CTBP2	One alternative transcript is a transcriptional repressor
36	cg13301327	No	rs11696871	ZBTB46	Zinc Finger, GO annotations include nucleic acid binding
37	cg13284789	No	rs4508746	SIDT1	GO annotations include RNA transmembrane transporter activity

To further prioritise prostate cancer relevant meQTLs, the presence of putative regulatory elements were examined 500 bp either side of the identified meQTLs. Again, the UCSC genome browser was used for this approach, specifically to determine the presence of regulatory elements such as transcription factor and miRNA binding sites, active histone and chromatin marks and DNase hypersensitivity regions. Concurrently considered, was the CpG density of each region, with regions containing too few CpGs (less than ten) excluded and regions containing dense CpG islands also excluded, as recent evidence suggests the greatest inter-individual variability occurs outside CpG islands, at shores and shelves. Utilising this filtering strategy, twenty-three CpGs were prioritised for further analysis and these are highlighted in orange in Table 4.3.

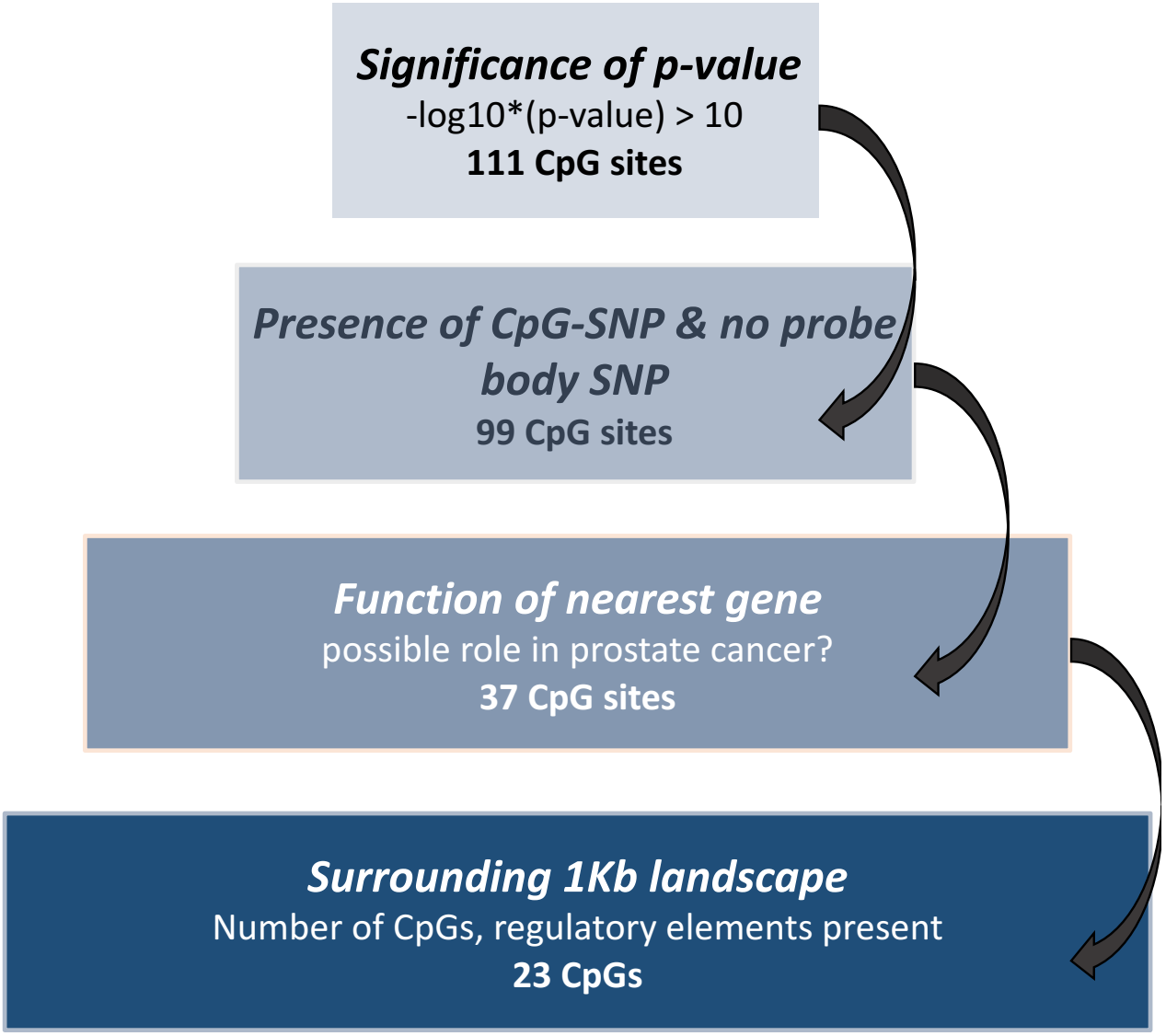


Figure 4.9 MeQTL filtering steps.
The five levels of filtering applied to the significant associations are presented together with the number of CpGs prioritised after each step.

4.5 Discussion

In order to identify inherited meQTLs associated with disease in this familial dataset, two distinct yet complementary approaches (Figure 4.2), were used. Following filtering as described, twenty-three meQTLs were prioritised for further follow up. These represent the most statistically significant associations identified, those in predicted regulatory regions and close to genes likely to be of functional relevance to prostate cancer.

Of the fifty significant associations with a $-\log_{10}$(p-value) greater than 10, there was a substantial enrichment of CpG-SNPs, with thirty-three meQTLs containing a CpG-SNP. This represented 68.8% of the most significant associations and is consistent with previous findings pertaining to the influential role of CpG-SNPs. For example, Zhi and colleagues found two thirds (or 66%) of the strongest meQTLs in their study were in fact CpG-SNPs (Zhi *et al.* 2013), while Shoemaker *et al.* observed 38-88% of allele-specific methylation occured at CpG-SNPs.

While the detection of significant associations between genotype and methylation is consistent with previous published studies, this does not necessarily indicate association with disease. The variability detected between individuals may simply be 'normal' epigenetic variability and not contribute to prostate cancer predisposition. To prioritise the most disease relevant CpGs, filtering was undertaken based on co-localisation to cancer-relevant genes. Of the thirty-seven meQTLs prioritised, seventeen have previously been linked to regulation of the cell cycle, apoptosis or cell growth and are therefore reasonable candidates to contribute to cancer. Nine

genes are putative tumour suppressor genes, eleven are involved in transcriptional activation or repression and nine have been associated with at least one form of cancer, with genes associated with multiple categories.

For the variable methylation approach, thirty-four CpGs identified as highly variable by the standard deviation method were not ranked in the top 100 variable sites identified by the 95% Reference Range. This included the most significant meQTL, cg13387643 as well as another nine of the top twenty most significant meQTLs identified from the standard deviation approach. This may be due to the fact that the 95% Reference Range excluded samples with very high or very low methylation levels as these were considered outliers, while the standard deviation approach included all methylation values. As it is difficult to discern whether outlying methylation values are technical errors or real biological signals, it is important at this stage to include all highly variable sites for subsequent biological analysis. Interestingly, of the sixty-seven CpGs that overlapped the two methods, fifteen were still prioritised in Stage.3 (twenty-seven CpGs remaining) and thirteen of the final twenty-three selected for validated were prioritised by both methods. These results support the use of a number of different statistical approaches for identifiying potential meQTLs.

Overall, the variable methylation approach identified the greatest number of highly significant associations with ninety-three for the standard deviation approach and eighty-five for the 95%-Reference Range approach. The design of the 450k array is focused on promoter and regulatory regions of the genome, and as such aligns with

the overall hypothesis of this study; that non-coding variation can trigger epigenetic changes in regulatory regions, leading to gene silencing, predisposing men to prostate cancer. It is therefore not surprising that the approaches aimed at variant prioritisation by initially examining methylation profiles generated more significant meQTLs than variants prioritised by other means. Additionally, the variable methylation approaches directly identify methylation patterns of interest; allele-specific methylation clusters, in three distinct methylation levels, driven by genotype.

Examination of the genetic regions previously associated with prostate cancer in familial studies revealed forty-eight of the prostate cancer risk windows with a statistical significant $-\log_{10}$(p-value) greater than 10. The windows associated with the risk loci through the Tasmanian Familial Prostate Cancer Study were selected for meQTL analysis as it was hypothesised that meQTL SNPs may have underpinned a portion of the linkage signal observed and thus be most relevant to this analysis of the same dataset. For thoroughness, risk loci identified in other published familial prostate cancer studies were also included in a second analysis. Heterogeneity in prostate cancer is well recognised and there is evidence that different sets of genes contribute to hereditary prostate cancer predisposition (Lange *et al.* 2003). The underpinning susceptibility genes for many of these large familial prostate cancer linkage regions remains unknown. Many of these linkage regions are large, spanning many megabases and it was hypothesised that identified non-coding meQTLs in these regions could underpin the prostate cancer risk. Comparison of the two sources of putative prostate cancer susceptibility loci revealed that within regions

surrounding the risk loci identified through the Tasmanian Familial Prostate Cancer Study, a greater proportion of significant and highly significant associations were identified, with 9% of associations remaining significant (after adjustment for multiple testing error). In comparison, 5% of loci identified through other published familial prostate cancer linkage regions generated significant associations. This is consistent with the hypothesis that inherited determinants of methylation changes are detectable in our dataset which may be associated with prostate cancer risk.

It is notable that for both of these analyses, the linkage regions were large, incurring substantial multiple testing correction. Further, the association method tests each SNP within each linkage region and thus assumes each has an equal likelihood of a potential association, whereas there is likely to be a single or few real associations in each region. Loci identified by modified linkage analysis of familial prostate cancer cases only tested for association in four genomic regions and this data was generated using the Affymetrix 370 SNP chip, a low density array compared with those available today, developed before much of the common or rare variation associations with disease had been discovered. Similarly, familial prostate cancer susceptibility loci identified from other published studies only examined a subset of the genome. It should also be considered that a proportion of the risk loci previously associated with prostate cancer may not be linked to meQTLs, instead affecting prostate cancer risk through other molecular mechanisms such as mutating the coding regions of genes or affecting other forms of gene regulation.

In contrast, the approach of using prostate cancer susceptibility SNPs identified by GWAS detected a large number of significant associations, with 55 in total. Each of the SNPs tested has been previously significantly associated with prostate cancer risk, further each of the SNPs tested were relatively common in the population, thus the higher success in identifying significant associations is not unexpected. Interestingly it could also indicate that susceptibility variants identified through GWAS are more likely to be acting to influence disease through altering methylation patterns. The majority of meQTLs identified in the current study were found in non-coding or intergenic regions. The evidence that a portion of prostate cancer risk SNPs identified through GWAS also contribute to familial prostate cancer risk (Jin *et al.* 2012; Teerlink *et al.* 2014), and that the fact non-coding variation consistently identified in GWAS may be underpinned by meQTls in non-coding regions (Barr and Misener 2016), suggests that a portion of the SNPs identified through this differential methylation approach may be located at loci previously identified as associated with prostate cancer susceptibility. Examination of this question revealed that three CpG sites overlapped in the list of significant sites generated through both the methylation and prostate cancer risk loci approaches (highlighted in yellow on Table 4.3) and these were selected to be taken forward in further analyses.

The observed influence of genotype on methylation levels at the meQTLs was also of interest here. As described previously and in section 4.3.3, methylation levels were found to cluster by genotype, as visualised on the smaller inset plot of Figure 4.7. It was also of interest to examine the effect of these meQTLs on neighbouring methylation patterns, as Smith *et al.* have demonstrated the effect of *cis*-meQTLs

extends up to 1500 Kb (Smith *et al.* 2014), with an earlier study observing the most significant meQTL associations for CpG-SNPs are within 45bp (Zhi *et al.* 2013). While Figure 4.8 does not show any perturbance in methylation patterns in the three CpGs either side of the variable CpG, this may be due to the limited coverage of the methylation array, as there are often large genomic distances between CpG sites represented on the array. Whilst this technology provides an excellent "snapshot" of the human methylome, it covers less than 2% of all genomic CpGs and as such, an alternative method of examining methylation patterns must also be utilised. A method such as bisulphite sequencing, capable of examining every CpG within a selected region, will allow the lack of variation surrounding meQTLs to be investigated in closer detail, aiding in the understanding of whether the uniform profiles observed are the result of a gap in analysis power on the methylation array, or a true biological phenomenon. In addition to allowing for a more complete picture of the methylation landscape surrounding meQTLs to be examined, bisulphite sequencing will provide an independent platform to validate the meQTLs of interest identified here.

While many studies now examine CpG-SNPs and meQTLs relative to cancer predisposition, at the commencement of this study there was a paucity of methodologies and analysis for such investigations, particularly for pedigree-structured datasets. As such, two innovative approaches were developed to examine the association between genotype, methylation and prostate cancer. Herein the genetic drivers of the most variable methylation sites, in conjunction with methylation surrounding genomic regions previously linked to prostate cancer risk

were examined and twenty-three regions were prioritized to be taken forward for validation and more detailed examination.

Chapter 5 – The influence of meQTLs on the surrounding epigenomic landscape and prostate cancer risk

5.1 Introduction

The role of CpG island function at promoters has been established for many years, with high levels of methylation at promoter islands consistently linked to gene silencing through establishment of a condensed chromatin state (Jones 2012). However, more recently attention has turned to the functional relevance of methylation at other regions of the genome. While gene body methylation has been less comprehensively studied, over the past two decades an understanding of the influence of gene body methylation on gene expression has emerged. Interestingly, gene body methylation has a contrasting effect on gene expression to that of promoter methylation, with low methylation linked to higher gene repression (Maunakea *et al.* 2010; Kulis *et al.* 2012; Varley *et al.* 2013). Yet, the function of gene body methylation remains to be fully understood, with recent studies only beginning to elucidate the underlying mechanisms controlling the effect of gene body methylation on gene expression (Yang *et al.* 2014).

Accompanying the growing interest in methylation at gene bodies, is a heightened appreciation for the role of methylation at shores and shelves, regions surrounding islands (see Chapter 1, Figure 1.4 for a schematic of a typical intergenic and genic landscape). This interest results from the observation that much of the variability

between tissue types and disease states occurs in these regions (Irizarry *et al.* 2009). As these regions are observed to be the sites of greatest variability between cancer and normal tissue, there is now considerable interest in both the molecular basis for this variability and also its functional consequences. One of the influences on these variable methylation patterns are meQTLs (methylation quantitative traits), which have been observed to alter methylation patterns across tissues and ancestry (Smith *et al.* 2014). MeQTLs are enriched in non-coding regions of the genome, where they impact on binding of transcription factors and chromatin remodellers (Heyn *et al.* 2014).

While a proportion of meQTLs contribute to disease burden, up to a third may simply contribute to natural human variation (Heyn *et al.* 2013) and the challenge is thefore to identify those that relate to disease. Much of the work in the epigenetic cancer field to date has focussed on examining the aberrant methylation changes that occur in cancerous tissue, and the distinctive global hypomethylation and regional specific hypermethylation observed in tumour tissues have been well documented (Jones 2012; Sproul and Meehan 2013). However, there has been limited analysis of inherited predisposing epigenetic factors that may be present throughout multiple tissues in the body, contributing to tumorigenesis. Rather than simply a consequence of the molecular dysregulation observed in tumours, if these aberrant methylation profiles are driven by inherited genetic variation, present in the germline and distributed throughout all tissues, the aberrant methylation profiles themselves may be contribute to cancer development (Hesson, Hitchins and Ward 2010). To investigate this potential role of meQTLs in driving prostate

cancer predisposition, the final aim of this study is to assess differential methylation levels at specific loci in peripheral blood of men affected by prostate cancer compared to unaffected individuals.

The greatest effect of meQTLs are often observed within close proximity to the genetic variant (Shoemaker *et al.* 2010; Zhi *et al.* 2013; Zhou *et al.* 2015) and therefore the methylation patterns immediately proximal to meQTLs are of interest. While the methylation array technology utilised to prioritise meQTLs in this study provides a valuable 'snapshot' across the genome, enabling disease-relevant regions of interest to be selected and prioritised, it captures only 2% of CpGs genome-wide (Fan *et al.* 2016), and therefore is not able to provide fine mapping methylation data. Bisulphite sequencing was therefore used to generate fine detailed mapping of methylation profiles in the regions proximal to selected meQTLs.

5.2 Methods

5.2.1. Sample Selection

DNA samples for bisulphite sequencing were selected from the Tasmanian Familial Prostate Cancer Resource. Samples analysed in Chapter 2 with high quality and quantity of DNA were selected, together with an additional twelve affected men from the familial resource. Thirty-seven of the familial samples (Appendix 5.1, highlighted in orange) were used to validate methylation data generated at earlier stages in this study and then further, to examine the influence of genotype on methylation profiles surrounding meQTLs. In addition, thirty-two age-matched

unaffected individuals were also selected from a second genetic resource, the Tasmanian Prostate Cancer Case Control Study, to compare methylation patterns between affected men from the familial resource (n=31) and unaffected population controls. A description of the sample set in this study can be found at (FitzGerald *et al.* 2009). As prostate cancer is markedly associated with age, unaffected men with the oldest age at sample collection were chosen for analysis. The unaffected men have been cross-checked with the Tasmanian Cancer Registry which records all cancers diagnosed in Tasmania, as required under legislation in Australia. The eighty-one individuals included in this analysis are listed in Appendix 5.1, together with their disease status, sex and age and median age for each of the unaffected and affected groups of individuals. To determine if such sample sizes would be sufficient to detect significant associations between epigenotype and cancer status, power calculations were conducted with the *pwr* R package. Various effect sizes (20%, 35%, 75%) were tested with a sample size of thirty samples per group. Various sample sizes (n=20-40 in each group) were then tested with the conservative effect size of 20% methylation difference. All tests were calculated for a significance level of 0.05, with all returning a power value above 0.999. These calculations thus indicate the sample size of thirty-two unaffected controls and thirty-one affected men would be sufficient to detect a significant difference between the groups.

5.2.2 PCR optimisation

Bisulphite converted DNA (see section 2.2.3 of Chapter.2 for details of bisulphite conversion) was diluted to 10ng/µL in the m-Elution Buffer provided in the EZ DNA *Methylation Gold Kit*™. Bisulphite PCR primers were designed in *methPrimer* (Li and

Dahiya 2002) to cover 1Kb regions of interest identified in Chapter 4. These primers either covered the entire region or for larger, more CpG dense regions, two overlapping fragments were selected to encompass the majority of CpG sites. See Appendix 5.2 for a list of the primers utilised in this analysis. Common SNPs were annotated to the converted reference sequences through the UCSC genome browser (Kent *et al.* 2002) and potential primer binding sites containing these SNPs were excluded.

Primers were first optimised using control placental DNA (Bioline $^{Pty\ Ltd}$), then conditions were tested on individual DNA samples of highest quantity/quality. Post bisulphite conversion, amplification was performed using a Veriti thermal cycler (Applied Biosystems). Optimisation of each primer pair was achieved +/- Q-solution (Qiagen) and/or Magnesium Chloride and a temperature gradient. In addition a variety of Taq polymerase mixtures (MyTaq^{TM} HS mix, Bioline$^{Pty\ Ltd}$ and GoTaq® Green master mix, Promega) were tested. See Appendix 5.3 for a list of final PCR conditions for each fragment. A subset of PCR amplified fragments were examined by gel electrophoresis on a 2% agarose gel to ensure amplification products were of expected size. Amplified fragments were stored at -80˚C prior to sequencing.

5.2.3 Nextera DNA and library preparation

DNA fragments from all regions were pooled for each individual, with a subset of the samples quantified on the Qubit® Flourometer (Invitrogen). Concentrations were adjusted as per manufacturer's instructions in preparation for fragmentation and tagging.

A 5µL volume of 0.2ng/µL genomic DNA per sample was used as the input for Illumina's Nextera XT Library Preparation kit. According to manufacturer's instructions, the DNA was 'tagmentated' (simultaneously fragmented and tagged with adapters), amplified through PCR and cleaned. Using a bead-based protocol, the libraries were normalised and pooled to ensure equivalent concentrations of each library was sequenced. Illumina's MiSeq was used to sequence the libraries.

5.2.4 Data generation, quality control and analysis

Standard quality control pipelines were employed to process next generation sequencing data. Low quality base calls and adapter fragments were removed using the adapter trimming tool *Cutadapt* (Martin 2011) with the wrapper *Trim Galore* (Krueger 2015). Quality scores on the trimmed data were then examined with *FastQC* (Andrews 2011) with high quality reads retained. Reads were aligned to a bisulphite Hg19 reference sequence using *Bismark* (Krueger and Andrews 2011) generating BAM and text files. Sequencing data was extracted from the BAM files with Bismark, and off target amplified regions removed, leaving 678 CpG sites in 13 genomic regions of interest. Data was then uploaded to R with the *BiSeq* package (Hebestreit and Klein 2013), and additional quality control thresholds were manually established using the mean number of CpGs covered per sample and the median coverage depth at these sites. Samples failing to reach a median coverage depth of ten reads across CpG sites were removed from further analysis, as were CpG sites exceeding a maximum quality threshold of 10% "no call" values across samples. Appendix 5.4 details the script used to generate the data and perform quality control.

5.3 Results

Using array-based analysis of samples from individuals sourced from the Tasmanian Familial Prostate Cancer meQTLs were identified and prioritised for further analysis (as detailed in Chapter 4). However, as the array based technology captures only 2% of the CpGs genome-wide, an additional approach was required to generate fine-mapping of the regions surrounding the meQTLs in order to determine the effect of the meQTL on the epigenetic profile of the surrounding region. Bisulphite sequencing was therefore performed on forty-nine peripheral blood samples from individuals in the Tasmanian Familial Prostate Cancer Study. To investigate the influence of these meQTLs on prostate cancer risk, samples from thirty-one of the affected men from the familial resource were then combined with an additional thirty-two samples from an independent resource, the Tasmanian Prostate Cancer Case Control Study. Due to time constraints, data generated from the analysis of twelve of the prioritised regions will be presented herein. These regions were those that were successfully amplified from bisulphite converted DNA and sequenced, as detailed in Table 5.1, which also provides genomic and epigenomic annotation data for these regions.

MeQTLs are located across a range of chromosomes and genomic locations, with variants located at either the cytosine or guanine of CpG pairs. This includes six C→T variants, five G→A variants and one C→A variants with the minor allele frequencies of these variants indicated in Table 5.1. Eight meQTLs are proximal to genes involved in regulation of cell growth, proliferation and migration, while four have frequently observed or putative tumour suppressor roles (Chapter 4, Table 3 has further detail

on the function of these genes). Six genes proximal to prioritised meQTLs have previously been associated with prostate cancer as indicated in Table 5.1.

Most of the regions examined appeared within gene introns (nine), while one represents an exonic region and two are located in untranslated regions (UTRs). Of the two meQTLs in UTRs, one is within 1500 bp of a transcription start site, as is another intronic meQTL. Relative to CpG island annotations, six meQTLs are located in “open seas” distant to islands while five are located in shores, regions flanking islands. Of note, none of these regions represent CpG islands, and contain between 11 and 51 CpGs.

Table 5.1 MeQTL regions successfully amplified by bisulphite sequencing

	Gene	Size *	Chr **	CpG number	RS number	Variant	Genotype Data	Proxy SNP available	MAF ***	CpG Annotation	Gene Annotation	Coding Annotation	CpGs in 1Kb	PC link+	++
1	*CASZ1*	930	2	cg13387643	rs284310	C --> T	No	Yes	T=0.30	OpenSea	Body	intron	29	YES	34
2	*ITGB2*	1217	21	cg02464073	rs1721	C --> T	Yes	NA	T=0.46	Shore	TSS1500	UTR	16	YES	27
3	*NME6*	751	3	cg08146865	rs3197223	C --> T	Yes	NA	T=0.18	OpenSea	3'UTR	UTR	11	NO	27
4	*C10orf46*	946	10	cg00231519	rs36101953	C --> T	No	Yes	T=0.26	Shore	TSS1500	intron	12	NO	24
5	*PRM1*	680	16	cg02978201	rs737008	C --> A	Yes	NA	A=0.49	OpenSea	Body	exon +++	27	NO	22
6	*FOXK2*	880	17	cg05331763	rs79974293	G --> A	No	No	A=0.03	Shore	Body	intron	33	NO	22
7	*USP7*	817	16	cg01891583	rs2304466	G --> A	Yes	NA	A=0.42	OpenSea	Body	intron	28	YES	22
8	*SEPT9*	1183	17	cg05161773	rs426439	C --> T	No	Yes	T=0.33	OpenSea	Body	intron	20	YES	21
9	*MGMT*	945	10	cg09993319	rs7898151	G --> A	No	Yes	G=0.43	OpenSea	Body	intron	21	YES	19
10	*RAB11B*	782	19	cg04610028	rs2967607	C --> T	No	No	T=0.36	Shore	Body	intron	51	NO	19
11	*AJAP1*	735	1	cg00345083	rs7517857	G --> A	Yes	NA	G=0.50	Shore	Body	intron	22	NO	15
12	*MCC*	1058	5	cg08238375	rs4705795	G --> A	No	Yes	A=0.43	OpenSea	Body	intron	13	YES	12

* PCR product size

** Chromosome Number

*** Minor allele frequency as established by the 1000 genomes project

+ Previously associated with prostate cancer

++ Significance as measured by -LOG10(p-value)

+++ Synonymous coding mutation

TSS1500: within 1500bp of the transcription start site

5'UTR: 5' Untranslated Region

NA: not applicable

5.3.1 Validation of methylation array data with bisulphite sequencing data

To validate methylation data generated earlier in this study (Chapters 2-4), values at the prioritised meQTLs between the methylation array and bisulphite sequencing data were compared. The absolute difference in methylation proportion (between 0-1, with 0 indicating no difference) between array and bisulphite sequencing values for sample PC22-16 were calculated at each meQTLs of interest. This sample had been separately interrogated across two batches on the methylation array. The average of the two values from each batch on the array was calculated at eleven of the twelve meQTLs (there was no sequencing data for the meQTL within the SEPT9 gene) and then subtracted from the average of the bisulphite sequencing values at corresponding meQTLs. The absolute difference between the two methylation array values as well as between the two platforms were plotted at each meQTL, as displayed in Figure 5.1 below.

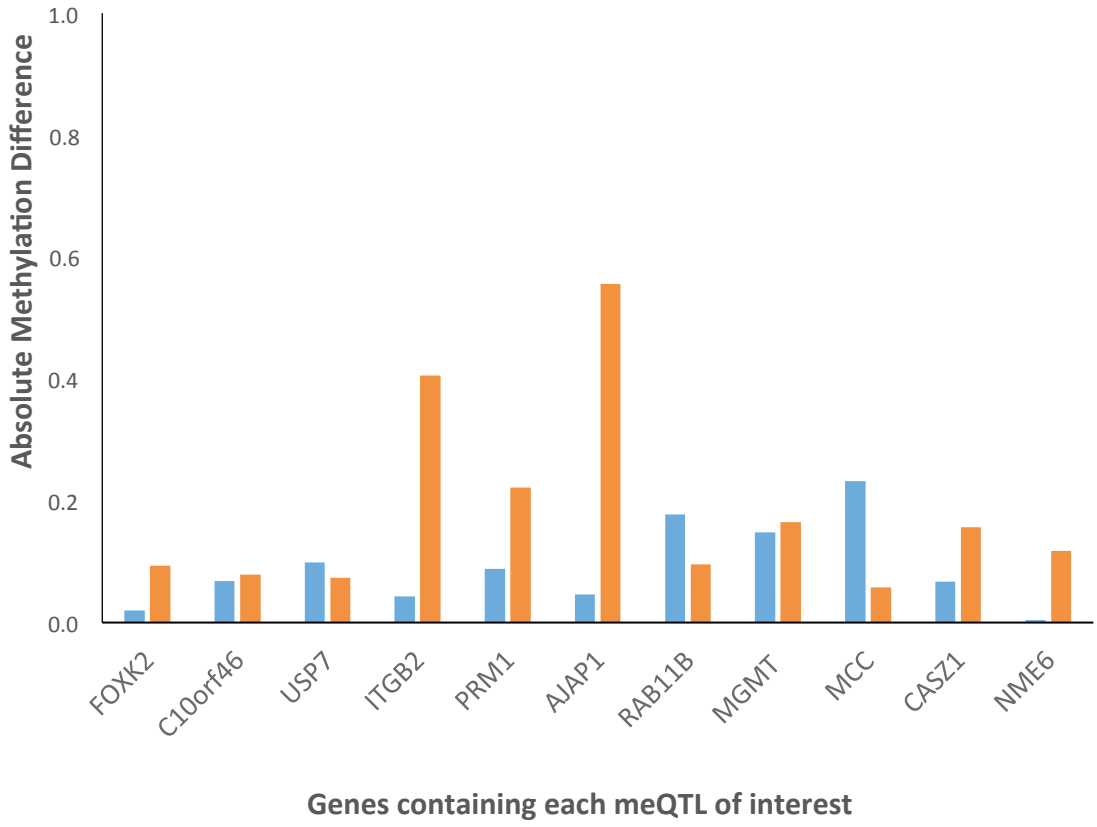


Figure 5.1 Absolute difference between methylation measures across meQTLs in sample PC22-16.
The absolute difference (0 representing identical values) between methylation values obtained from the array and bisulphite sequencing data are presented in red bars across 11 of the meQTLs, while blue bars represent the difference between technical replicates of sample PC22-16 on different array batches.

Nine of the meQTLs had differences lower than 0.3 between the two platforms (red bars), indicative of a consistent biologically relevant methylation pattern between the platforms. This is within the validation range reported by (Bibikova *et al.* 2011). Namely, if genotype drives methylation profiles in allele-specific clusters, as hypothesised in this study, then ten out of twelve meQTLs for this sample have consistent biological methylation patterns across the platforms. Furthermore, as highlighted by the blue bars in Figure 5.1, the two technical replicates generated using the methylation array did not produce identical results, with three meQTLs demonstrating a greater difference within the array platform than between platforms. These differences highlight the variability often present within the same platform, demonstrating the need to examine methylation profiles by multiple methods.

An additional comparison of the two datasets was performed by comparing the differences between methylation values at each CpG across forty-three samples interrogated on both platforms. The absolute difference at each CpG value per sample was calculated then averaged across the samples per CpG. The absolute mean difference for each meQTL is plotted in Figure 5.2 below, together with error bars indicating the standard error of the mean. Excluding *SEPT9*, which had a much lower number of successfully sequenced samples at the meQTL (n=24), the differences between the technologies were consistent across the meQTL regions, with discrepancies between the platforms possibly the result of the heightened sensitivity in bisulphite sequencing.

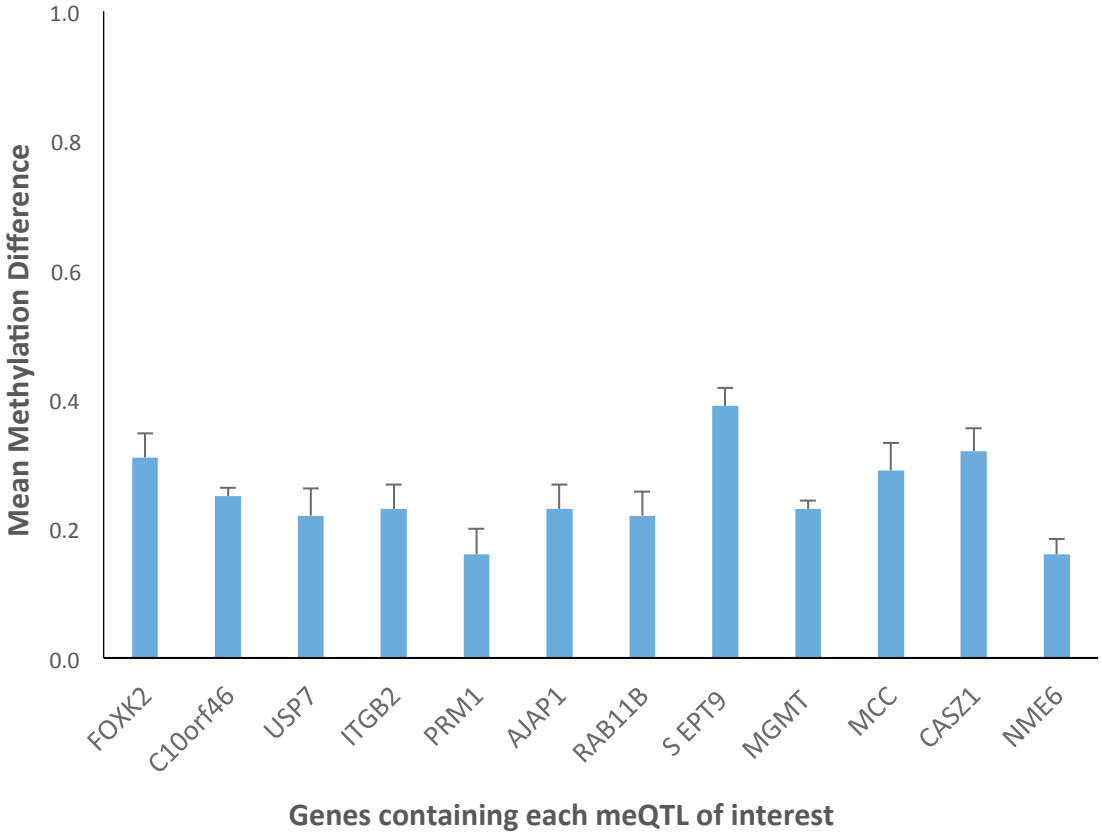


Figure 5.2 Mean absolute differences between array and bisulphite sequencing data.

The mean absolute difference (0 representing identical values) between methylation values obtained from the array and bisulphite sequencing data are presented across 12 genes containing meQTLs of interest. 43 samples were sequenced for each meQTL, with successful data generated for 35-43 samples at 11 meQTLs. *SEPT9* was the exception, with lowever sample coverage (n=24). Error bars are indicative of the standard error of the mean.

5.3.2 Exploring the influence of meQTLs on the methylation landscape.

To examine the influence of genotype on methylation patterns in *cis*, samples drawn from the familial resource were divided into three groups based on genotype at the CpG-SNP. SNPs were either located at the cytosine or guanine of CpG pairs. Only samples with genotype data (n=37) were examined here. For five meQTLs, the CpG-SNP was included in the genotype data, and as such genotypes were directly extracted using the *GenABEL* R package. Genotypes at the CpG-SNP were also confirmed through whole genome sequencing (WGS) data for a subset of samples (n=15) across all genotypes. For the remaining seven meQTLs, the CpG-SNP itself was not present in the genotype data, (see Table 5.1 for details on which meQTLs had genotype data and which could be imputed). For five of these meQTLs, genotype at the CpG-SNP was imputed from neighbouring proxy SNPs in linkage disequilibrium (LD). The online tool *SNAP* (Johnson *et al.* 2008) was used for imputation. Data was drawn from the Northern European population of the Hapmap project. Following imputation Four of the selected SNPs had R^2 values greater that 0.95 and a D' of 1, and the fifth had R^2 0.83 and D' 1. For the remaining two SNPs, no appropriate tagging SNP could be imputed. Thus genotype data was available for ten meQTLs. The remaining two meQTLS for which genotype data was not available were not considered further.

Ten meQTL regions were qualitatively analysed for genetically driven methylation patterns, by examining the median methylation per genotype group at each CpG within the amplified 1Kb region surrounding the meQTL. All meQTL regions displayed distinct methylation clusters corresponding to genotype. To quantitatively examine

the effect of genotype on methylation in these regions, the meQTL with the largest effect size (CASZ.1, as determined by the qualitative analysis) was tested with a linear model. This meQTL was not significant and it was concluded that the sample size was not large enough to detect significance at any of the meQTLs. The effect sizes for the meQTLs are presented in Figures 5.3 and 5.4 and can be used for power analyses to determine the sample size needed to detect significance in subsequent studies.

To examine the influence of meQTLs on prostate cancer predisposition, methylation at the regions of interest were compared between affected individuals (n=31) selected from the Tasmanian Familial Prostate Cancer Resource and age-matched unaffected individuals (n=32) drawn from the Tasmanian Prostate Cancer Case Control Study. The median methylation level within each group was calculated at every CpG per meQTL region and plotted across the genomic region.

5.3.3 Genetic variation driving aberrant methylation profiles: A proof of principle at the meQTL proximal to *CASZ1*

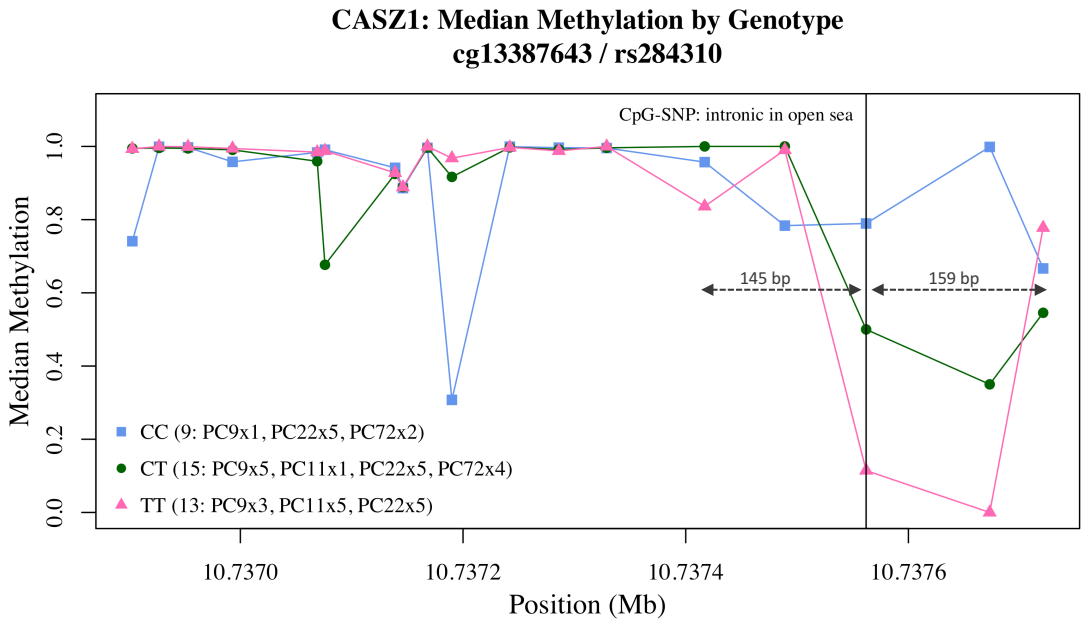
The meQTL associated with the CASZ1 gene, was located in a intron within the gene body. Methylation status at this CpG-SNP was strongly influenced by genotype, with individuals of the CC genotype displaying high levels of methylation (median methylation proportion 0.79), while those with TT genotype display low methylation (median methylation proportion 0.11) and heterozygous individuals display 50% methylation (median methylation proportion 0.50). Genotype groups exhibited

expected methylation levels, as individuals with a TT genotype lack the potential for cytosine methylation in the CpG context. The familial distribution of genotype frequencies was examined and displayed on Figure 5.3A, with no family contributing overly towards one certain genotype.

The influence of the *CASZ1* meQTL on genotype is not restricted to the CpG-SNP itself but extends to at least one neighbouring CpG either side, corresponding to approximately 150 bp either side of the CpG-SNP, as indicated on Figure 5.3A. As data is not displayed for every CpG in the region, due to failure to reach quality control thresholds, there may be a greater influence of the CpG-SNP on surrounding patterns than evident here.

A similar differential methylation profile was also observed between affected and unaffected individuals at the *CASZ1* regions, as indicated in Figure 5.2B, where a substantial difference between affected and unaffected individuals was evident at the CpG-SNP. Affected individuals have lower median methylation, at 0.18, similar to the 'TT' genotype in 5.2A at 0.11, while the unaffected individuals display mid-level methylation at 0.63, similar to heterozygous allele-specific methylation in Figure 5.2A. Strikingly, the extended influence of the meQTL seen in Figure 5.2A, can also be observed in Figure 5.2B, where the affected individuals have lower methylation either side of the CpG-SNP than the unaffected individuals. Intriguingly, epigenetic dysregulation of this gene has been linked to neuroblastoma and prostate cancer.

A)



B)

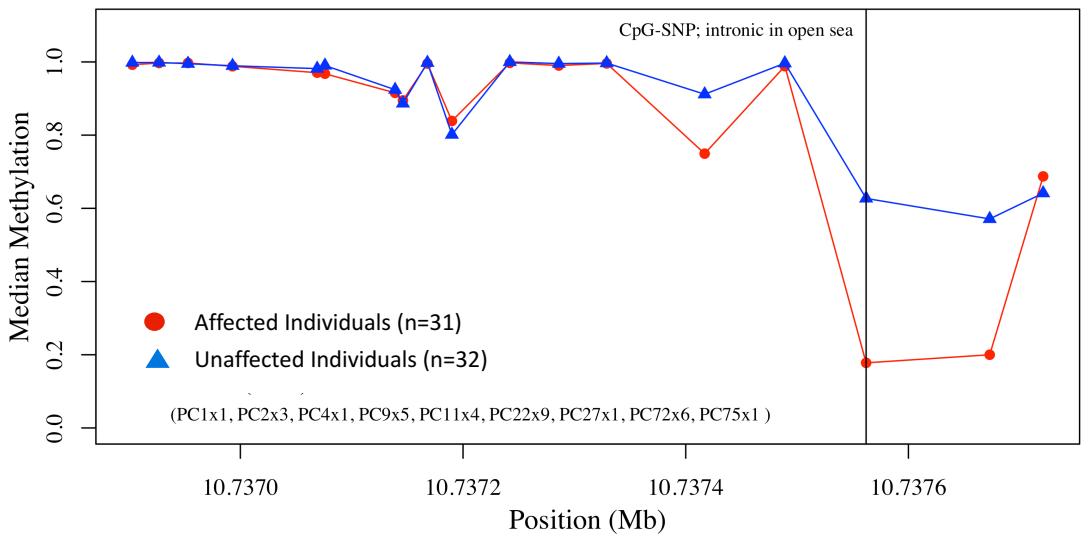


Figure 5.3 Median methylation profiles surrounding the meQTL of interest
Median methylation is plotted against corresponding genomic positions for 18 CpGs in the *CASZ1* region. A) Samples (n=37) are divided into genotype groups, as indicated in the key, with median methylation per group plotted at each CpG. A black vertical line represents the location of the CpG-SNP, with the corresponding genomic and CpG annotation indicated adjacent to the line. The dashed lines indicate the extent of the influence of the CpG-SNP on neighbouring CpG methylation levels. B) Median methylation for unaffected individuals (n=32) and affected individuals (n=31), with familial distributions of the affected individuals indicated.

5.3.4 Extension of methylation profile analysis to other meQTL regions

The remaining nine meQTL regions analysed for genetically driven methylation patterns (as detailed in Figure 5.4 (A, C, E, G, I, K, M, O, Q), all displayed distinct methylation clusters corresponding to genotype. In the differential analysis by prostate cancer status, seven showed divergent patterns between affected and unaffected individuals at the CpG-SNP (Figure 5.3 B, D, F, H, J, P, R) while two (Figure 5.3 L, N) had similar median methylation levels between the two groups. Of the seven regions with dissimilar profiles, six had lower methylation in the affected individuals with only one (F) displaying lower methylation in the unaffected individuals. The methylation profiles between the genotype stratified plots and the disease status plots at each meQTL were similar for all regions except two (Figure 5.3 K&L, M&N) which showed clear genotype clustering at the CpG-SNP, yet minimal differences between affected and unaffected individuals at the same site.

Specifically, the two meQTL regions within UTRs (Figure 5.4 A&B at the *ITGB2* gene and Figure 5.4 A&B at the *NME6* gene) displayed similar methylation profiles, with high methylation levels seen throughout the region and interrupted only by methylation changes at the CpG-SNP. MeQTLs located in intronic open sea regions (*USP7* Figure 5.4 I&J*, MGMT* Figure 5.4 M&N*, MCC* Figure 5.4 Q&R) displayed minimal variation in methylation profiles between individuals, except at the CpG-SNPs. The region annotated to *AJAP* (Figure 5.4 O&J) was CpG dense and displayed variable methylation levels over the region, which were most divergent at the CpG-SNP. This variability was expected as the region is located in and intronic island

shore. The sole region located in an exon at *PRM1* (Figure 5.4 G&H), was tightly clustered between individuals with distinct variation between individuals only present at the CpG-SNP.

Interestingly, the region annotated to the *CDK2 associated cullin domain 1* (*C10orf46 or CACUL1*) gene shown in Figure 5.4 E&F, is located approximately 150bp downstream of a CpG island. While methylation patterns at the CpG-SNP were dependent on genotype and prostate cancer status, the region further upstream and more proximal to the CpG island was more uniformly methylated between all genotype groups and between individuals with different cancer status. This region displayed lower methylation levels, as often seen at CpG islands located at promoter regions. Surprisingly, the region annotated to *MGMT,* a gene well known to display aberrant epigenetic dysfunction in cancer and indeed used as a treatment biomarker in certain types of brain cancer (Wiewrodt *et al.* 2007), did not show any differences between affected and unaffected individuals at the CpG-SNP (Figure 5.4 N), despite a genotype driven methylation pattern at the CpG-SNP (Figure 5.4A).

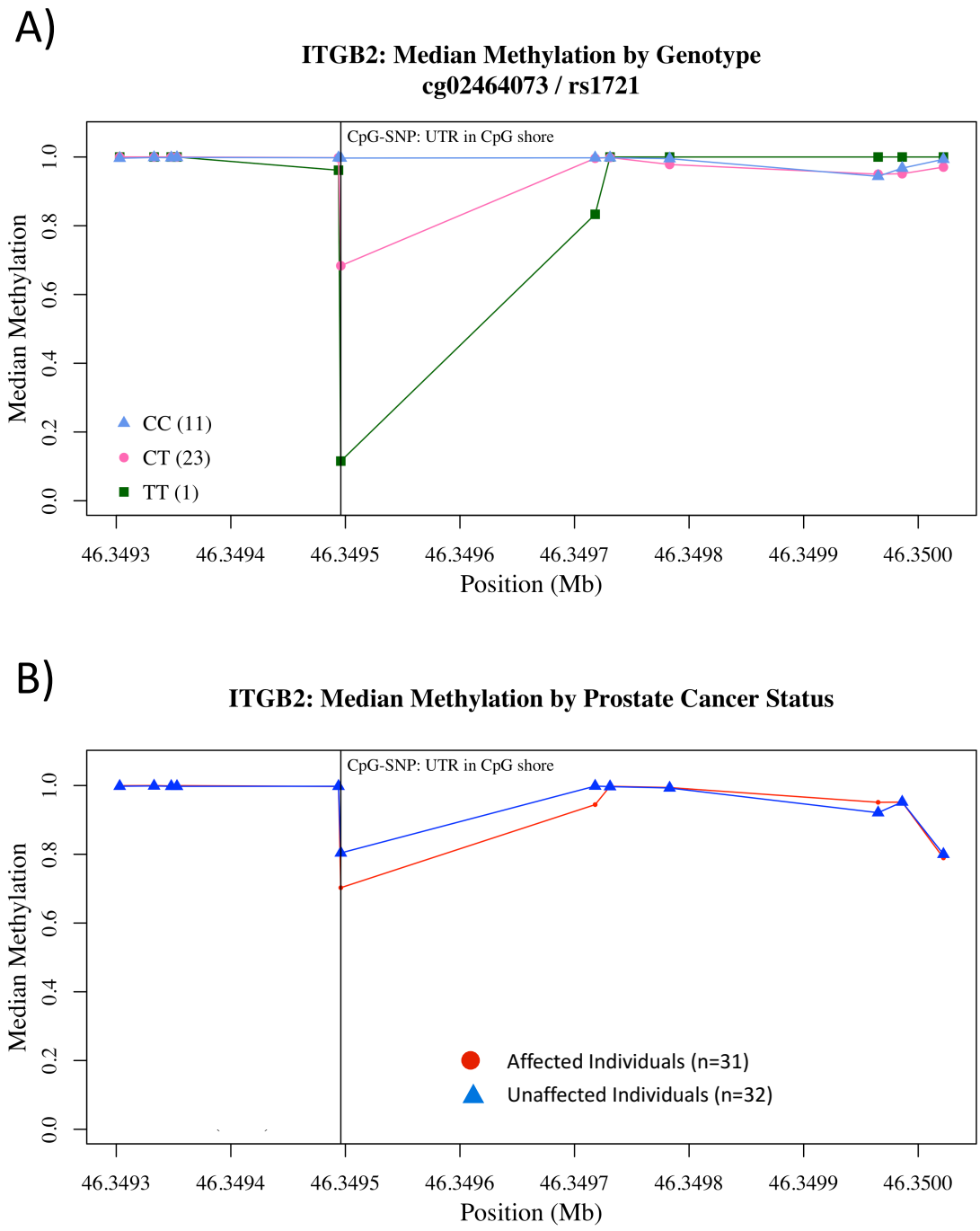
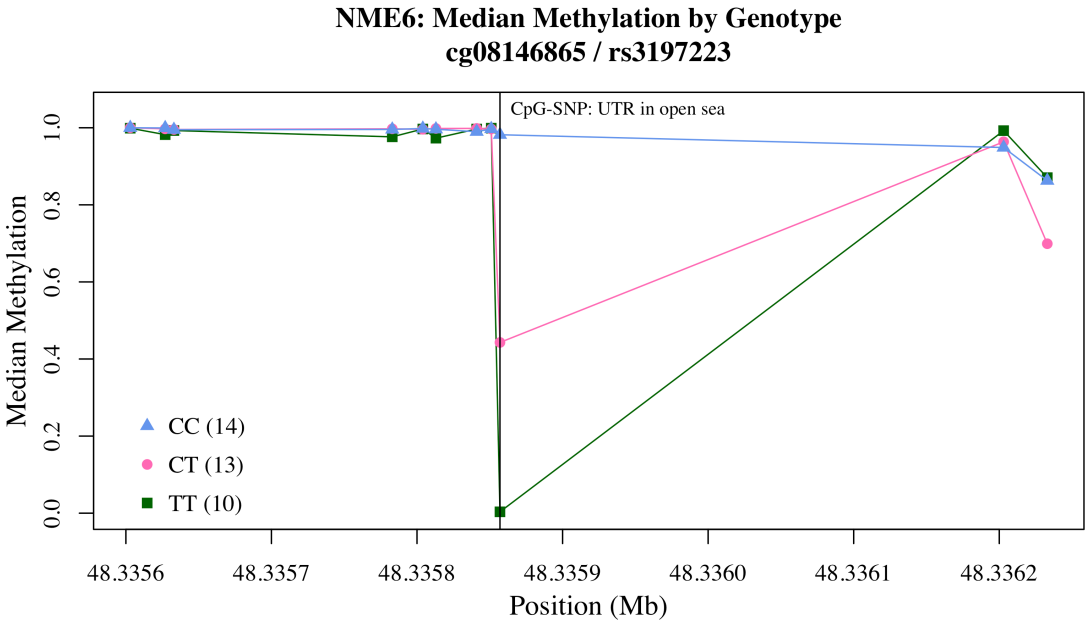


Figure 5.4 A & B Median methylation profiles surrounding the meQTL of interest in *ITGB2*.

Median methylation for 12 CpGs in the *ITGB2* region. A) Samples (n=37) are divided into genotype groups, as indicated in the key, with median methylation per group plotted at each CpG. A black vertical line represents the location of the CpG-SNP, with the corresponding genomic and CpG annotation indicated adjacent to the line. B) Median methylation for unaffected individuals (n=32) and affected individuals (n=31).

C)



D)

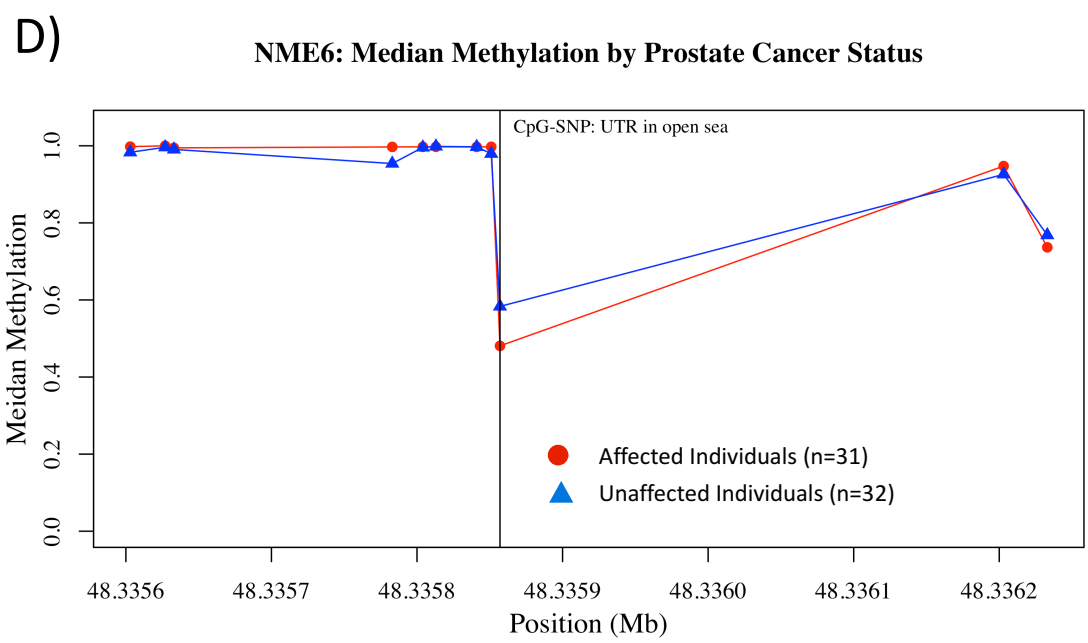
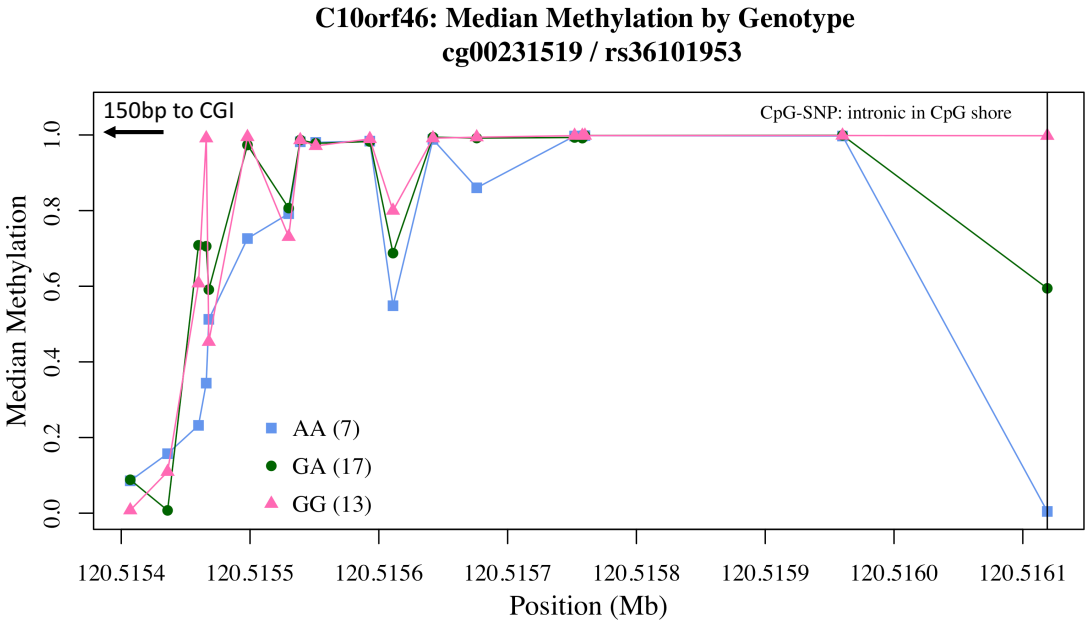


Figure 5.4 C & D Median methylation profiles surrounding the meQTL of interest in *NME6.*

Median methylation for 11 CpGs in the *NME6* region. A) Samples (n=37) are divided into genotype groups, as indicated in the key, with median methylation per group plotted at each CpG. A black vertical line represents the location of the CpG-SNP, with the corresponding genomic and CpG annotation indicated adjacent to the line. B) Median methylation for unaffected individuals (n=32) and affected individuals (n=31).

E)



F)

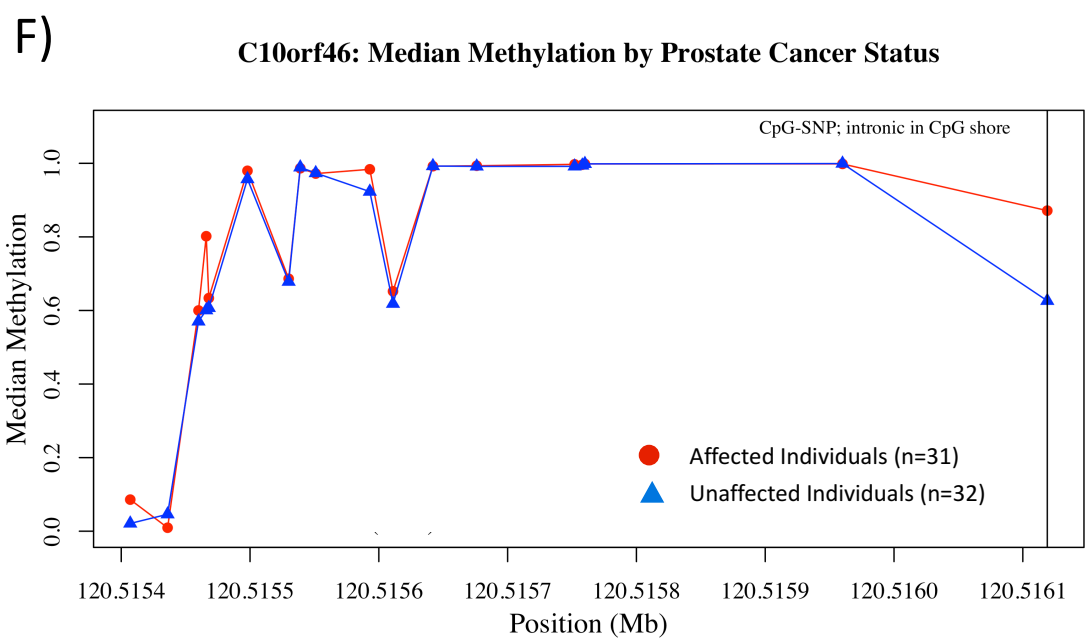
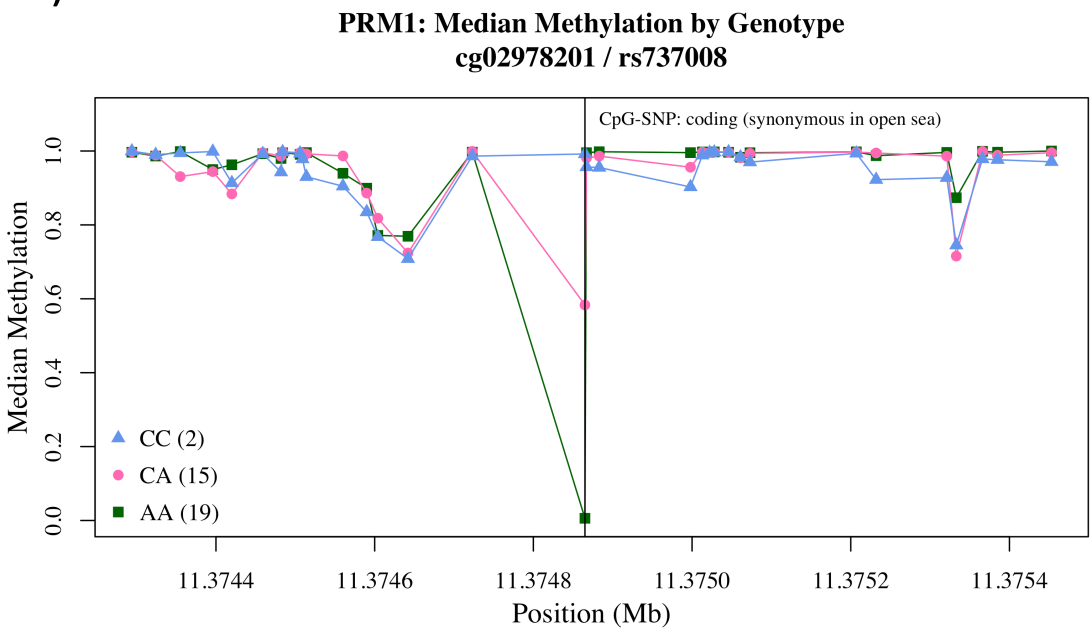


Figure 5.4 E & F Median methylation profiles surrounding the meQTL of interest in *C10orf46*.

Median methylation for 18 CpGs in the *C10orf46* region. A) Samples (n=37) are divided into genotype groups, as indicated in the key, with median methylation per group plotted at each CpG. A black vertical line represents the location of the CpG-SNP, with the corresponding genomic and CpG annotation indicated adjacent to the line. The black indicates a CpG island is 150bp upstream. B) Median methylation for unaffected individuals (n=32) and affected individuals (n=31).

G)



H)

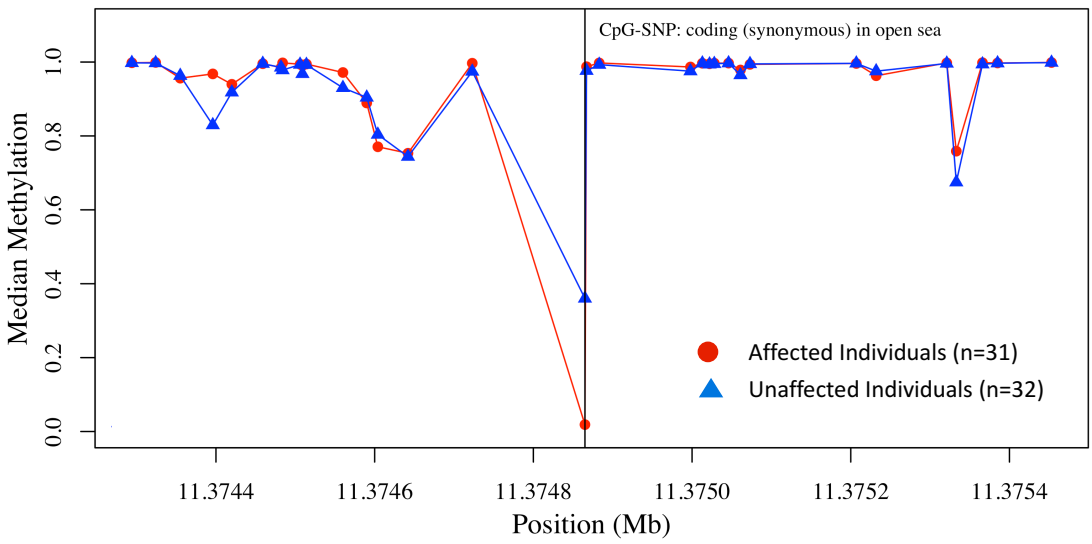
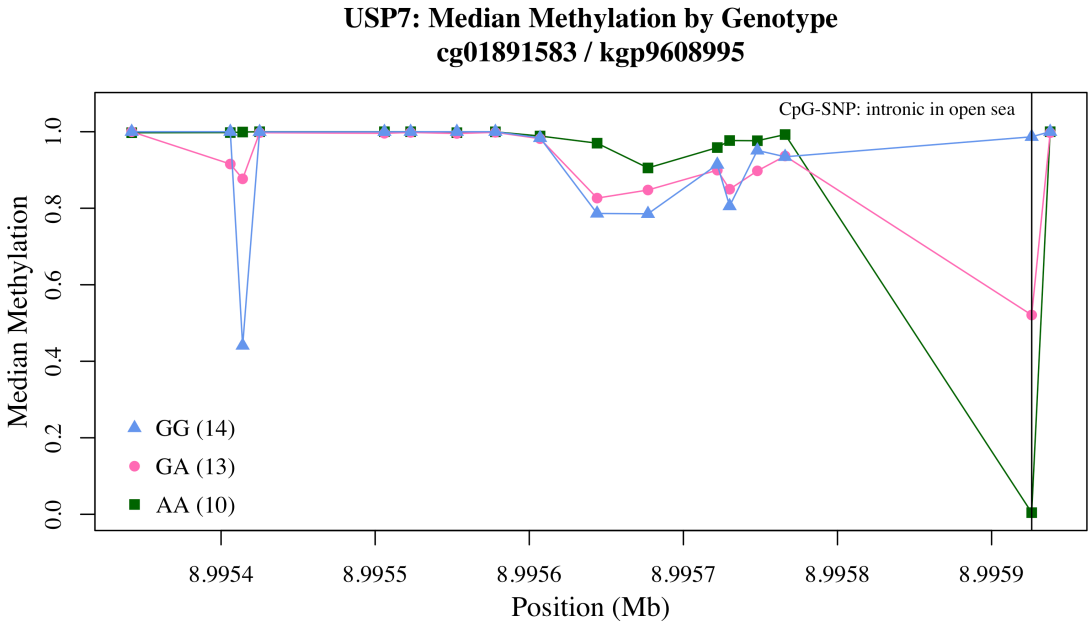


Figure 5.4 G & H Median methylation profiles surrounding the meQTL of interest in *PRM1*.

Median methylation for 33 CpGs in the *PRM1* region. A) Samples (n=37) are divided into genotype groups, as indicated in the key, with median methylation per group plotted at each CpG. A black vertical line represents the location of the CpG-SNP, with the corresponding genomic and CpG annotation indicated adjacent to the line. B) Median methylation for unaffected individuals (n=32) and affected individuals (n=31).

I)



J)

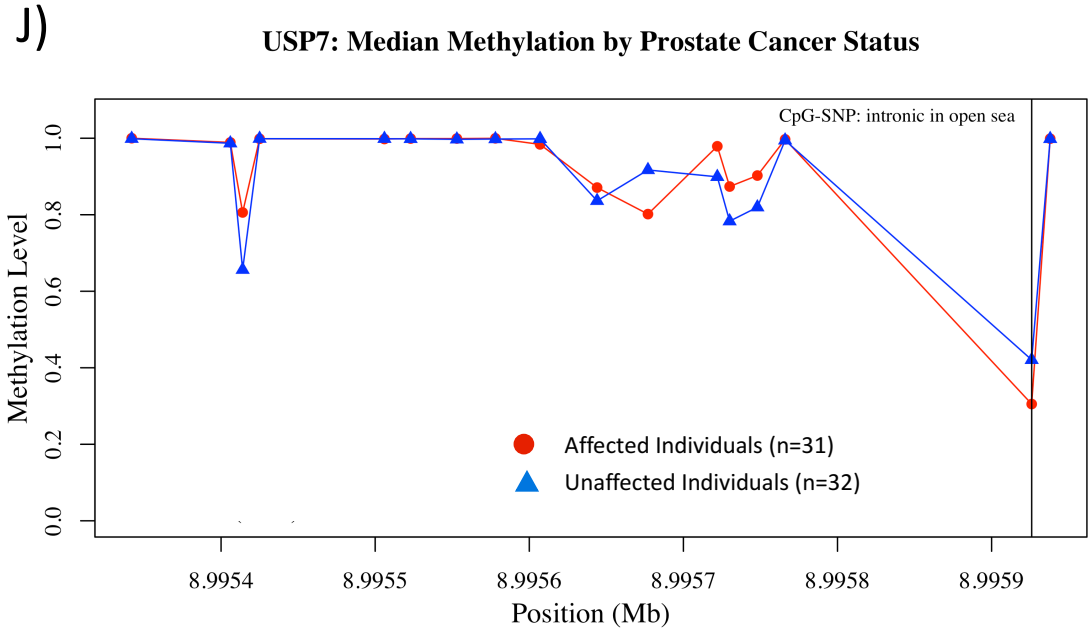
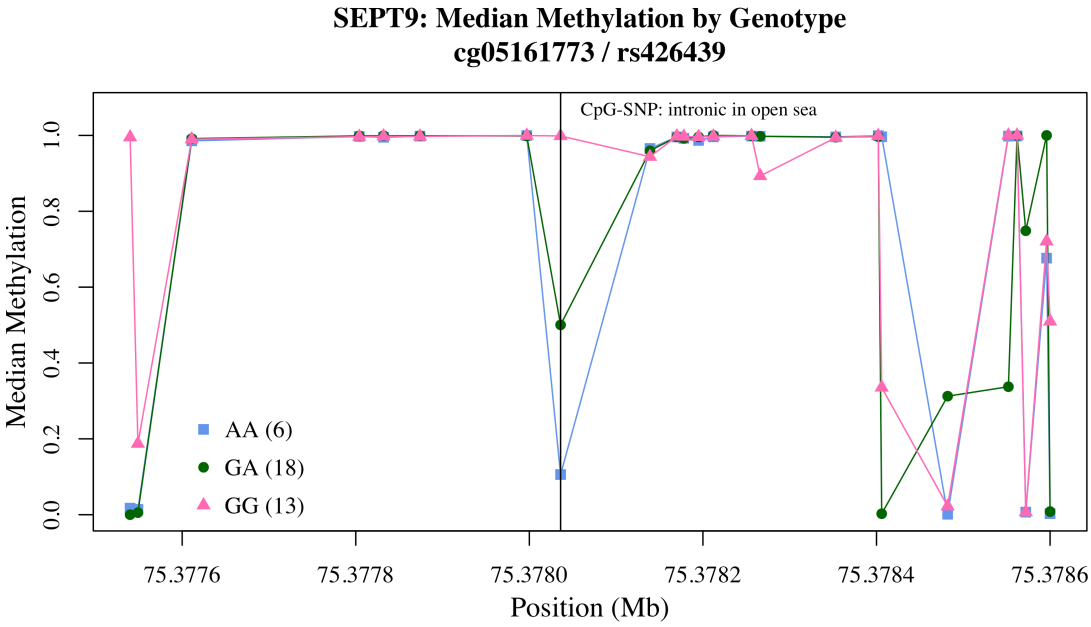


Figure 5.4 I & J Median methylation profiles surrounding the meQTL of interest in *USP7*

Median methylation for 17 CpGs in the *USP7* region. A) Samples (n=37) are divided into genotype groups, as indicated in the key, with median methylation per group plotted at each CpG. A black vertical line represents the location of the CpG-SNP, with the corresponding genomic and CpG annotation indicated adjacent to the line. B) Median methylation for unaffected individuals (n=32) and affected individuals (n=31).

K)



L)

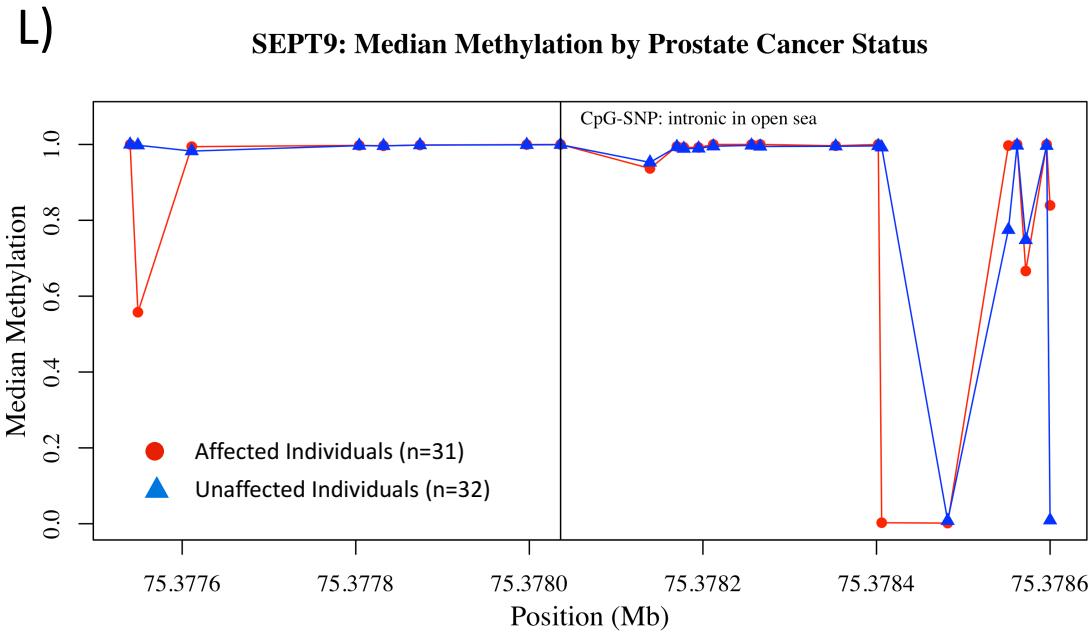
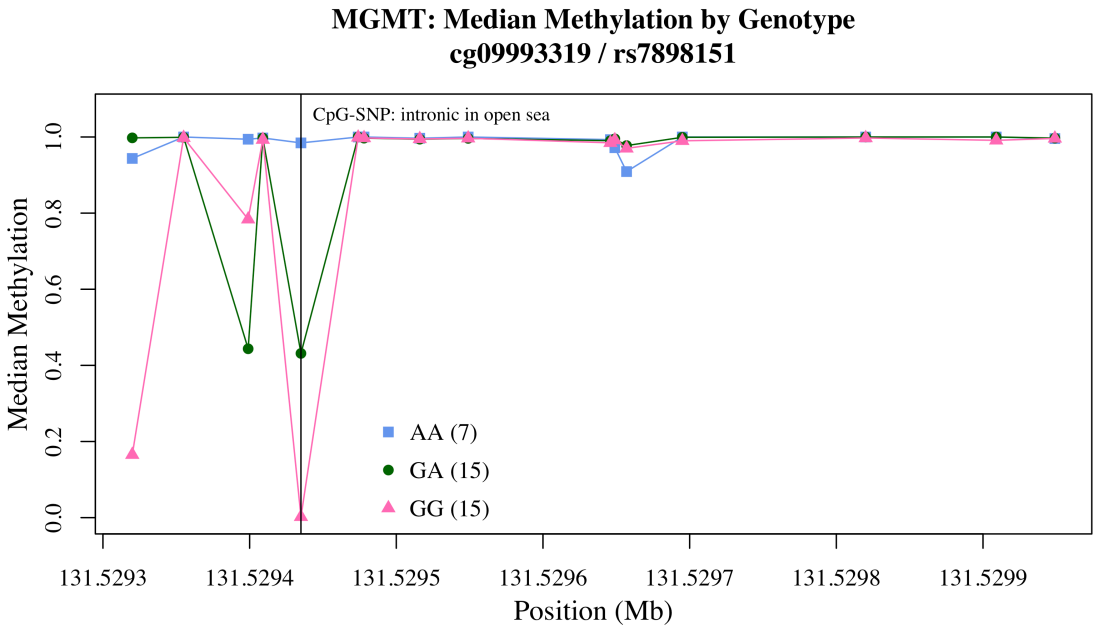


Figure 5.4 K & L Median methylation profiles surrounding the meQTL of interest in *SEPT9.*

Median methylationfor 24 CpGs in the *SEPT9* region. A) Samples (n=37) are divided into genotype groups, as indicated in the key, with median methylation per group plotted at each CpG. A black vertical line represents the location of the CpG-SNP, with the corresponding genomic and CpG annotation indicated adjacent to the line. B) Median methylation for unaffected individuals (n=32) and affected individuals (n=31).

M)



N)

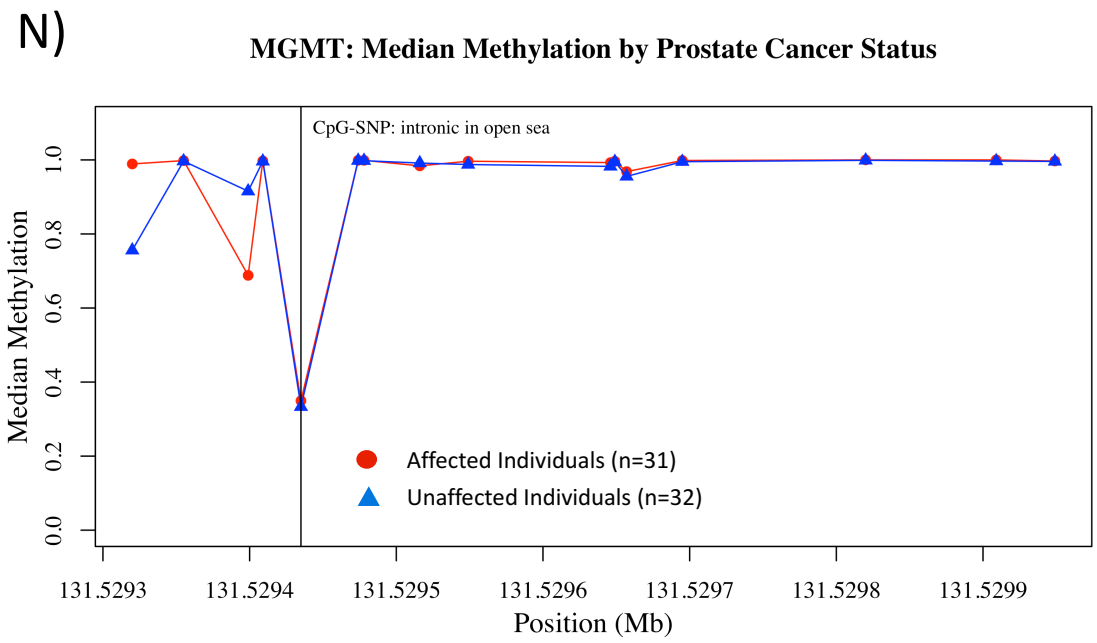
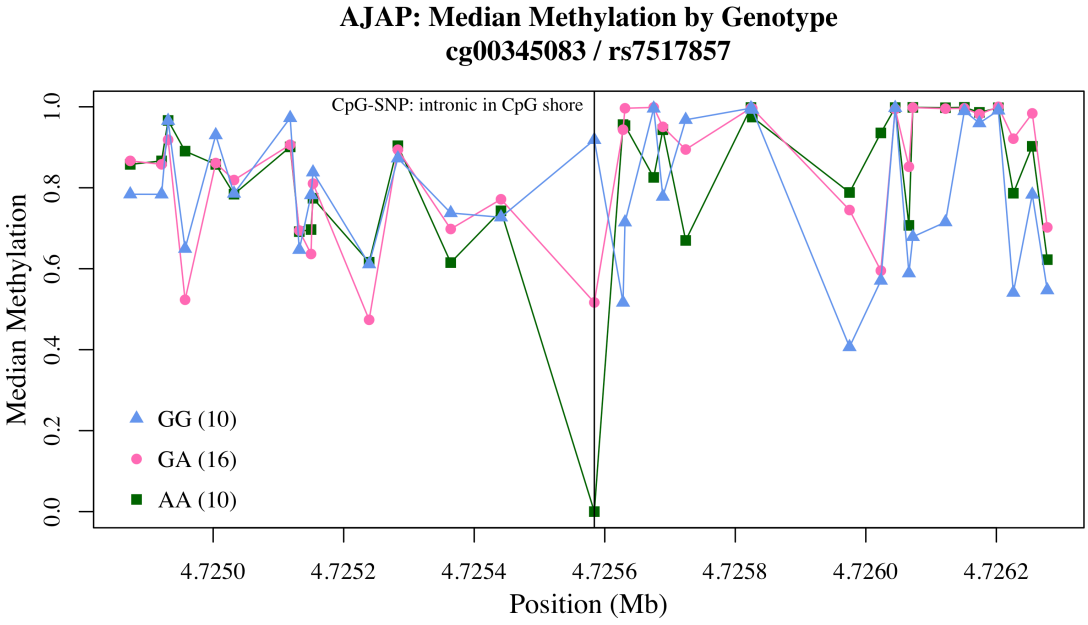


Figure 5.4 M & N Median methylation profiles surrounding the meQTL of interest in *MGMT*.

Median methylation for 16 CpGs in the *MGMT* region. A) Samples (n=37) are divided into genotype groups, as indicated in the key, with median methylation per group plotted at each CpG. A black vertical line represents the location of the CpG-SNP, with the corresponding genomic and CpG annotation indicated adjacent to the line. B) Median methylation for unaffected individuals (n=32) and affected individuals (n=31).

O)



P)

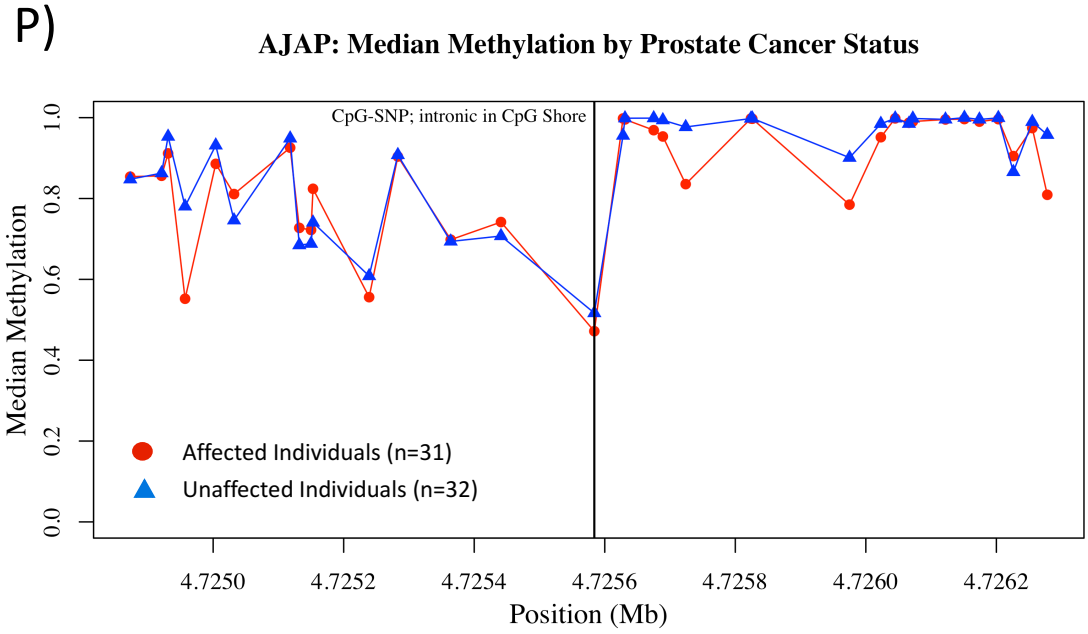
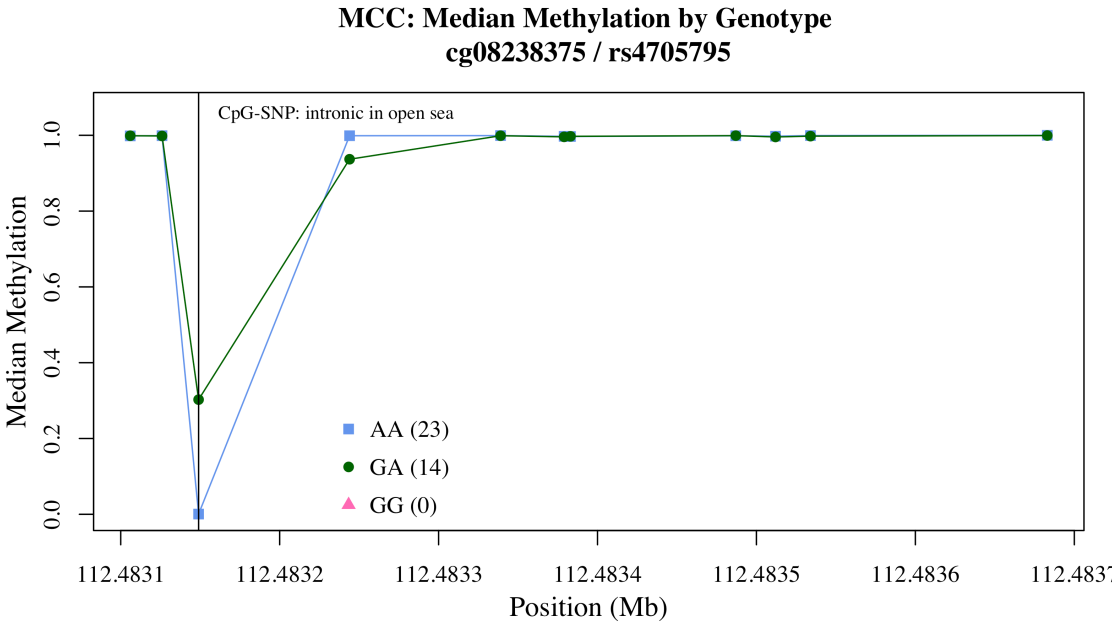


Figure 5.4 O & J Median methylation profiles surrounding the meQTL of interest in *AJAP1.*

Median methylation for 34 CpGs in the *AJAP1* region. A) Samples (n=37) are divided into genotype groups, as indicated in the key, with median methylation per group plotted at each CpG. A black vertical line represents the location of the CpG-SNP, with the corresponding genomic and CpG annotation indicated adjacent to the line. B) Median methylation for unaffected individuals (n=32) and affected individuals (n=31).

Q)



R)

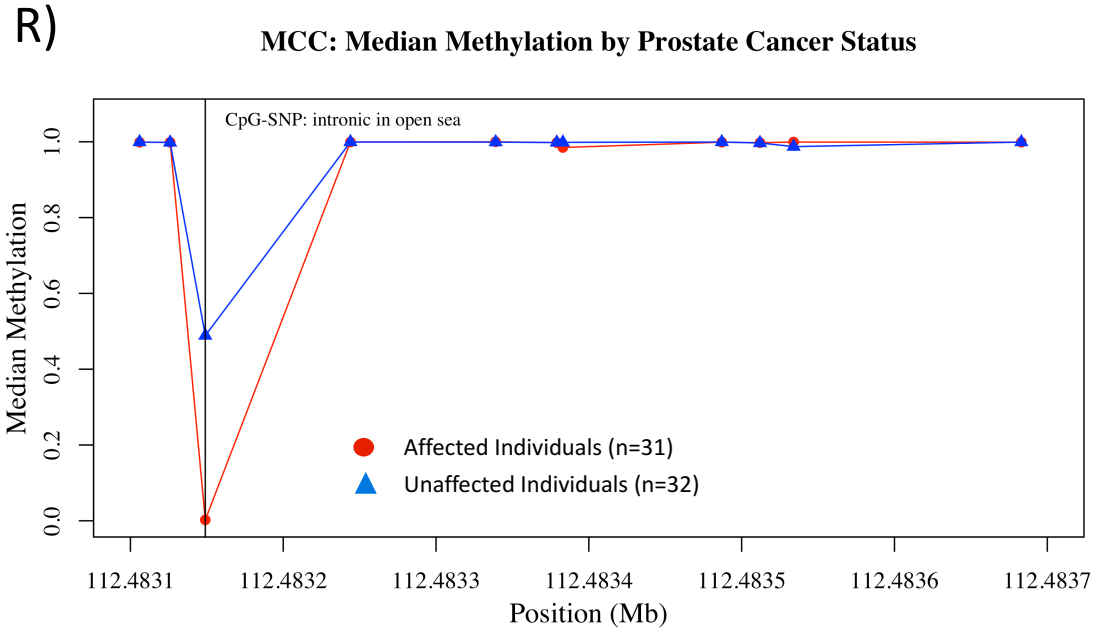


Figure 5.4 Q & R Median methylation profiles surrounding the meQTL of interest in *MCC*.

Median methylationfor 11 CpGs in the *MCC* region. A) Samples (n=37) are divided into genotype groups, as indicated in the key, with median methylation per group plotted at each CpG. A black vertical line represents the location of the CpG-SNP, with the corresponding genomic and CpG annotation indicated adjacent to the line. B) Median methylation for unaffected individuals (n=32) and affected individuals (n=31).

5.4 Discussion

The vast majority of disease risk loci identified for complex disease to date are in non-coding regions where the functional effect on the underlying mechanisms of predisposition are difficult to elucidate (Hindorff *et al.* 2009). This is particularly the case for prostate cancer, where the regulatory mechanisms surrounding many identified risk SNPs are yet to be fully understood (Whitington *et al.* 2016). While these variants can affect a range of regulatory mechanisms, including non-coding RNA (ncRNA) (Jin *et al.* 2011; Kumar *et al.* 2013; Ramalho-Carvalho *et al.* 2016) and histone modifications (Esteller 2007), this study has focussed on examining the effect of non-coding variation on the well-established epigenetic regulatory mechanism; DNA methylation.

Altered methylation profiles are a molecular hall-mark of cancer (Jones and Baylin 2007; Jones 2014), with numerous aberrant methylation profiles observed at key genes associated with prostate cancer progression, including the cell growth regulatory gene *G1 To S Phase Transition 1 (GSTP1)* and the mismatch repair gene *O^6-methylguanine DNA methyltransferase (MGMT)* (Kang *et al.* 2004; Ellinger *et al.* 2008). However, as previously discussed, these observed changes often occur well into the tumorigenesis process, where it is difficult to distinguish driver from passenger epi-mutations. Family studies have emerged as a powerful tool to overcome this challenge, as driver mutations can be more readily characterised if a variant can traced through a family pedigree, where genotypes can be compared between multiple affected and unaffected relatives over many generations (Williams and Blangero 1999).

As such, the current study examines genetically driven methylation patterns in individuals drawn from a rare familial resource. These pedigrees are likely to be enriched for deleterious inherited drivers of prostate cancer, as the resource contains dense aggregates of affected men, spanning several generations. MeQTLs were identified and prioritised using array-based techniques (Chapter 2-4), with the independent method of bisulphite sequencing used here to validate this data and further explore the methylation landscape surrounding these variants. Twelve meQTLs were validated, with the correlation between genotype and methylation in related individuals examined. It is hypothesised that meQTLs in non-coding regions drive these aberrant methylation patterns, interacting with other regulatory mechanisms to bring about permanent silencing of tumour suppressor genes, predisposing men to prostate cancer.

The effect of methylation on gene expression is dependent on the location and distribution of CpG sites throughout the genome, with promoter methylation associated with active gene expression and gene body methylation associated with the opposite (Jones 1999). Thus, the location of CpGs relative to genes and CpG islands is crucial to understanding the function of CpG methylation in both health and disease states. Accordingly, the genomic (for example whether intergenic, intronic, exonic) and epigenomic (relation to CpG island) annotations at prioritised meQTLs were carefully considered during the current analysis and are detailed in Table 5.1. Seven of the twelve meQTLs were in "open sea" regions, representing isolated CpGs, located further than 4Kb from CpG islands (Sandoval *et al.* 2014). It is

not surprising that these regions constituted the majority of identified CpG locations, as the greatest proportion of annotated CpGs on the array are located in "open seas" (36%). The remaining five meQTLs were located in CpG island shores, which are 2Kb either side of islands, comprising 23% of annotated CpGs on the array. None of the twelve meQTLs were located in CpG islands as only regions with lower CpG density were prioritised for follow up in Chapter 4.

The prioritised meQTLs were enriched in regulatory regions, with nine of the twelve prioritised meQTLs located in introns, two in untranslated regions and only one in an exon. While the methylation array has been developed with a focus on regulatory regions, it is highly enriched for promoter regions, with intergenic and promoter CpG sites comprising 45% of regions on the array (Dedeurwaerder *et al.* 2011). Thus meQTLs may be naturally enriched outside of islands, at CpG shores, shelves and open seas, and may explain why promoter regions were not prioritised in the pipeline of Chapter 4. This is supported by the methylation profile at one region, annotated to *C10orf46* (Figure 5.4 E&F), where the start of the region, located approximately 150bp downstream of a CpG island exhibited more uniform methylation patterns between individuals than upstream at the CpG-SNP. This lower variability in methylation is often seen at CpG islands located at promoter regions, with Irizarry and colleagues observing island shores to be the most variable CpG regions during development and across multiple cancer types (Irizarry *et al.* 2009).

The association between epigenotype and prostate cancer occurrence was then examined in the affected individuals drawn from the familial resource and

unaffected individuals from an independent population-based study. While seven of the twelve genes containing prioritised meQTLs have previously been associated with prostate cancer, none of the exact risk SNPs have previously been linked to the disease. As these samples were drawn from families enriched for prostate cancer cases, these novel variants may help to clarify part the unexplained inherited component of prostate cancer risk.

The most striking difference in gene body methylation between genotype groups and prostate cancer disease status was observed at the *CASZ1* gene, a zinc finger transcription factor involved in neuronal cell differentiation, with a putative tumour suppressor role (Liu *et al.* 2006). At the CpG-SNP a TT genotype was associated with minimal methylation. This was expected, as a C→T mutation removes the potential for cytosine methylation. This differential methylation profile was mirrored in the disease status plot, where samples from men with prostate cancer had minimal methylation levels compared to unaffected individuals. Both these differential methylation patterns extended approximately 150 bp either side of the CpG-SNP. While reduced gene body methylation has been previously observed to result in a decrease in gene expression (Yang *et al.* 2014), to date, no studies have examined the correlation between gene body methylation and gene expression. However, *CASZ1* expression has been found to be decreased in aggressive neuroblastoma cells, with the ectopic restoration of *CASZ1* expression inhibiting tumour cell growth, both *in vitro* and *in vivo* (Carén *et al.* 2007; Liu *et al.* 2011).

While *CASZ1* is most widely known to be involved in neuronal cell differentiation, with the majority of cancer studies examining aberrant *CASZ1* expression and function in brain cancer, there is often overlap between genes involved in various cancers (Amundadottir *et al.* 2004). Indeed, this may occur with *CASZ1*, as lower *CASZ1* expression was recently associated with increased prostate cancer cell growth *in silico* (Chiyomaru *et al.* 2012). This study found several genes including *CASZ1*, were targets of the oncogenic micro RNA, miR-151. MiR-151 is highly expressed in prostate cancer cell lines, with downregulation through the isoflavone genistein, found to supress prostate cancer cell growth in PC3 and DU145 cell lines (Chiyomaru *et al.* 2012). To date the direct effects of *CASZ1* expression on prostate cancer cell lines or *in vivo* have yet to be examined.

The underlying mechanism for the involvement of *CASZ1* in prostate cancer or neuroblastoma is also yet to be understood, with no tumour-associated variation in the coding region of the gene identified (Wang *et al.* 2012). As such it has been suggested that epigenetic silencing may be involved, with downregulation of the histone methyltransferase *EZH2* associated with an increase in *CASZ1* expression and a decrease in neuroblastoma growth (Wang *et al.* 2012). The role of methylation in this regulatory process is less clear, as while DNA methylation has been examined at several CpG islands associated within *CASZ1* regulatory regions, no difference was observed between primary tumour DNA and the blood of unaffected individuals (Carén *et al.* 2007). It may be that alterations in expression of *CASZ1* are driven by aberrant methylation patterns outside islands, as detailed in this study.

Intriguingly, a rare sub group of prostate cancers (approximately 1%) possess a neuroendocrine phenotype (also termed small-cell carcinoma), which are more likely to develop resistance to androgen therapy (Deorah *et al.* 2012) (Grigore, Ben-Jacob and Farach-Carson 2015). Normal prostate epithelium is in fact interspersed with neuroendocrine cells. These cells function to produce specialized peptides to stimulate the release of normal prostate secretions. Inappropriate regulation of these cells can trigger cellular proliferation, angiogenesis and migration of prostate cells (Bok and Small 2002). As *CASZ1* is transcription factor involved in neuronal development and observed to be dysregulated in both neuroblastoma and prostate cancer. This represents and interesting avenue for further research through investigation of whether dysregulation of this gene could in turn disrupt the regulation of neuroendocrine cells and contribute to prostate cancer development. Indeed, data from a recent study of castration-resistant neuroendocrine prostate cancer indicates up to 36% of this prostate cancer phenotype displays alterations in *CASZ1* either through amplification, upregulation or deep deletion (Beltran *et al.* 2016).

This study has validated twelve of the prioritized meQTLs (identified in Chapter 4) and further examined the methylation landscape surrounding a subset of these meQTLs. All ten regions examined for genetically driven methylation patterns, had genotype dependent methylation profiles at the CpG-SNP, which extended in *cis* at the *CASZ1* meQTL region. Men affected by prostate cancer had lower median methylation levels at seven of the nine CpG-SNP sites, with one region, at the *CASZ1* gene, also demonstrating lower methylation either side of the CpG-SNP. This CpG-

SNP may be informative in further understanding the complex inherited component of prostate cancer as, decreased expression of the gene has been previously associated *in silico* with increased prostate cancer cell growth (Chiyomaru *et al.* 2012). Further experimental analysis of this region is required to help elucidate the potential molecular mechanisms involved. Data generated in this study has uncovered a large number of meQTLs potentially involved in prostate cancer risk, with twelve validated and examined in further detail here and one providing an elegant proof of principle of the prioritisation pipeline. However, there remains a wealth of information to be further examined, both bioinformatically and experimentally in future studies.

Chapter 6 – Conclusions

Prostate cancer is one of the most prevalent cancers world-wide, yet significant clinical challenges in diagnosis and treatment persist. Diagnostic difficulties are exacerbated by a limited understanding of underlying molecular mechanisms governing the development and progression of the prostate cancer. This is exemplified by our ongoing reliance on the biomarker, PSA which has limited sensitivity and specificity. Of pressing clinical concern is our inability to differentiate men at high risk of metastasis from those with a more indolent form of the disease. This is exceedingly pertinent for men diagnosed with prostate cancer, as while current treatments can have substantial negative impacts on quality of life, early treatment is critical for men with an aggressive form of the disease, as once prostate cancer has metastasised there is currently no cure, and limited treatment options.

Prostate cancer is one of the most heritable cancers (Hjelmborg *et al.* 2014), yet the majority of genetic drivers remain to be elucidated, with only a third of predicted variants thus far detected (Olama *et al.* 2014). A key to unravelling the underlying molecular mechanisms of prostate cancer, may therefore be a more detailed understanding of the underlying inherited component of the disease. One way to unravel this "missing heritability" is through pedigree-structured studies, which provide a powerful tool to exploring genetic predisposition to many complex diseases. This is partly due to the ability to study rare variants, which are often enriched in families with high rates of certain diseases, together with diminished

genetic noise. The additional knowledge about relatedness of samples and inheritance patterns of genetic markers allows other computational advantages such as imputation (Saad and Wijsman 2014).

Familial studies have proven successful in identifying novel risk regions and variants in prostate cancer, some of which have been validated in larger population studies, and importantly have been applicable to men outside of the familial disease risk population (Teerlink *et al.* 2014; Kote-Jarai *et al.* 2015). The success of familial studies in unravelling the inherited component of complexes diseases has been particularly evident in Tasmania, where the population has been relatively stable since the 1800s, with the majority of the current populace descended from a founder population of settlers and convicts from the UK. Examining extensive pedigrees from this population, where many large families can be identified and traced back to their common founders, has enabled a further understanding of disease aetiology in complex disorders such as glaucoma (Fingert *et al.* 1999), Huntington's disease (Brothers 1964) and cancers including multiple endocrine neoplasia (Burgess *et al.* 2000) and more recently, prostate cancer (Eeles *et al.* 2009).

Over 93% of disease-associated SNPs identified to date are in non-coding regions of the genome (Hindorff *et al.* 2009; Kumar *et al.* 2013), and focus has therefore more recently turned to investigation of the role of non-coding variants in gene regulation and complex disease aetiology (Barr and Misener 2016) , which has been enabled by recent advances in next generation molecular technologies.

The central hypothesis of the current study was that DNA sequence changes in non-coding regions of the genome can alter transcriptional activity and trigger epigenetic changes in regulatory regions, leading to gene silencing and contributing to prostate cancer development and progression. Two broad aims were examined to investigate this hypothesis. Firstly, to use the Tasmanian Familial Prostate Cancer Resource to examine the association between genotype, as measured by SNPs, and epigenotype, as measured by DNA methylation. Secondly, to examine the association between epigenotype and prostate cancer occurrence.

The major findings of this study encompass the development of an analysis pipeline for pre-processing familial methylation data and performing meQTL with an emphasis on identifying meQTLs linked to prostate cancer risk. Through this pipeline, the influence of genotype on methylation profiles and subsequently these methylation profiles on prostate cancer risk were able to be examined. One meQTL region, at the *CASZ1* gene, exhibited genotype dependent methylation profiles in *cis* to the meQTL, with the altered methylation profile also observed in men affected by prostate cancer, when compared to unaffected controls. As epigenetic dysregulation has previously been linked to neuroblastoma and prostate cancer it acts as a "proof of principle" for the pipeline. Numerous other meQTL regions of interest requiring further follow up were also identified, however these are outside the scope of this study.

Limited methodologies existed for meQTL analysis of familial data at the commencement of this study, with a major outcome of the study being the establishment of a quality control, pre-processing and analysis pipeline for meQTL data generated with a familial study design. This pipeline, together with recommendations for future methylation array-based studies utilising familial data are described in Chapter 3 Figure 10 and published in (Cazaly *et al.* 2016). This pipeline is not only relevant for familial studies with the same design as the current study, but can also be applicable to a range of varied study designs, including longitudinal studies. Longitudinal studies examining differential methylation between twins at various time points or epigenetic changes within the same individual over time, for example those associated with age; also lack two distinct experimental groups on which to perform normalisation. Additionally, the methylation differences within the same individual over time may be much subtler than differences between cancerous and normal tissue and thus require pre-processing methods robust at removing technical bias. As such, these studies would greatly benefit from the pipeline developed here.

The pipeline enabled prioritization of prostate cancer risk meQTLs (Chapter 4). After validation on an independent platform (targeted bisulphite sequencing, Chapter 5), the methylation landscape surrounding twelve of these meQTLs was mapped in finer detail. Nine meQTL regions exhibited genotype dependent methylation patterns at the CpG-SNP.

The association between epigenotype and prostate cancer risk was then examined by comparing the methylation profiles between affected men drawn from the familial resource and age-matched unaffected controls selected from an independent case-control study (enabled through targeted bisulphite sequencing, Chapter 5). Men affected by prostate cancer displaying lower median methylation levels at eight of the twelve prioritized CpG-SNP sites.

One meQTL region, encompassing the *CASZ1* gene, demonstrating genotype dependent methylation levels at the CpG-SNP as well as in *cis* either side. There was also lower methylation in the men affected by prostate cancer than in unaffected controls, with the pattern extending approximately 150bp either side of the CpG-SNP. As changes in expression of this gene have been previously associated with neuroblastoma *in vitro* and *in vivo* (Carén *et al.* 2007; Liu *et al.* 2011) and with increased prostate cancer cell growth *in silico* (Chiyomaru *et al.* 2012), the meQTL may be informative in further understanding the complex inherited component of prostate cancer. Specifically, it would be important to investigate whether the DNA methylation patterns observed in peripheral blood in this study extend to aberrant patterns in prostate tissue. Subsequent investigations would examine whether the methylation differences do affect gene expression, and if differential expression is observed between unaffected controls and affected individuals. Finally, the effect of ectopically altering regions of the *CASZ1* gene via a gene editing technique such as *CRISPR/CAS9* could be examined. The rapid development of such gene editing techniques (Mali *et al.* 2013; Cong *et al.* 2013) now facilitates the introduction of specific mutations into cells and the examination of the effects of such mutations on

cells. This technology is being widely used to introduce cancer associated mutations in *vitro* and in *vivo* studies to examine effects on tumour development. As such, this region provides a "proof of principal" for the pipeline developed in this study, as it aimed to select meQTLs with a potential role in prostate cancer predisposition which could then be validated experimentally.

At the commencement of the current study, one of the few examples of the influence of meQTLs on cancer predisposition came from the study of Lynch syndrome, often involving mutations to DNA mismatch repair genes (Ward *et al.* 2013). A proportion of these cases are attributed to 'secondary' epimutations, or aberrant methylation patterns resulting from underlying sequence changes such as promoter deletions and SNPs (Hitchins and Lynch 2014).

During later stages of the current study, the publication of additional studies have further demonstrated the utility of employing familial resources to examine the effect of genetic drivers on epigenetic patterns, particularly at methylation quantitative trait loci (meQTLs) (Lemire *et al.* 2015; Kulkarni *et al.* 2015). Kulkarni and colleagues examined epigenome-wide methylation in 850 Mexican-American's from the San Antonio Family Heart Study, noting methylation at 14% of CpGs was associated with DNA sequence variation in *cis* (Kulkarni *et al.* 2015). A subset of this methylation was also associated with type 2 diabetes phenotypic traits, including alterations at five well characterised diabetes-associated genes. Such meQTL mapping may provide insight into the underlying molecular mechanisms of diabetes and aid in potential treatment targets or diagnostic techniques. Examining

more distal influences of meQTLs, Lemire *et al.* profiled 898 colon cancer patients and 850 controls from the Ontario Familial Colon Cancer Registry and reported 1,919 trans-meQTLs, of which 90% replicated in at least one independent data set. Many of these *trans*-meQTLs are associated with epigenetic regulators, such as long non-coding transcripts and zinc-finger transcription factors, which can then influence epigenetic patterns on health and disease (Lemire *et al.* 2015).

Genetic variation can affect epigenetic patterns and gene expression through numerous interconnected mechanisms, including DNA methylation, non-coding RNAs, histone modifications and transcription factor binding (Furey and Sethupathy 2013; Ramalho-Carvalho *et al.* 2016; Whitington *et al.* 2016). Advances in technology have enabled development of the extensive ENCODE database (ENCODE Project Consortium *et al.* 2012), which aims to catalogue all functional elements in the human genome. The current study has focussed on one aspect of epigenetic regulation- DNA methylation. Specifically, 5-methylcytosine has been examined, and while the role of 5-hydroxymethylcytosine is becoming prominent in the epigenetics field, particularly in studies of the brain (Kriaucionis and Heintz 2009), the mark is beyond the scope of the current study. Additionally, the effect of meQTLs in this study was only examined in *cis*, at regions proximal to the meQTLs themselves. It is frequently observed that these variants can have much further influences on gene regulation in *trans* (Lemire *et al.* 2015) however, again, these aspects of gene regulation are outside the scope of the current study. This study provides insight into one aspect of epigenetic regulation, by developing an analysis strategy for investigating non-coding variants that can be applied to other familial studies and

diseases. The reality of the epigenetic landscape is far more complex, comprising many varied mechanisms working in concert to facilitate gene regulation. A comprehensive understanding of complex disease predisposition requires a more extensive understanding of the interactions between these mechanisms.

References

Aguilera O, Fernandez AF, Muñoz A *et al.* Epigenetics and environment: a complex relationship. *J Appl Physiol* 2010;**109**:243–51.

Allis CD. Beyond the Double Helix: Reading and Writing the Histone Code. *FASEB J* 2008;**22**.

Amundadottir LT, Thorvaldsson S, Gudbjartsson DF *et al.* Cancer as a Complex Phenotype: Pattern of Cancer Distribution within and beyond the Nuclear Family. Lewis C (ed.). *PLOS Med* 2004;**1**:e65.

Anway MD, Cupp AS, Uzumcu M *et al.* Epigenetic transgenerational actions of endocrine disruptors and male fertility. *Science* 2005;**308**:1466–9.

Anway MD, Skinner MK. Transgenerational effects of the endocrine disruptor vinclozolin on the prostate transcriptome and adult onset disease. *Prostate* 2008;**68**:517–29.

Aryee MJ, Jaffe AE, Corrada-Bravo H *et al.* Minfi: a flexible and comprehensive Bioconductor package for the analysis of Infinium DNA methylation microarrays. *Bioinformatics* 2014;**30**:1363–9.

Aryee MJ, Liu W, Engelmann JC *et al.* DNA methylation alterations exhibit intraindividual stability and interindividual heterogeneity in prostate cancer metastases. *Science Translational Medicine* 2013;**5**:169ra10–0.

Auton A, Durbin RM, Bentley DR *et al.* A global reference for human genetic variation. *Nature* 2015;**526**:68–74.

Banovich NE, Lan X, McVicker G *et al.* Methylation QTLs Are Associated with Coordinated Changes in Transcription Factor Binding, Histone Modifications, and Gene Expression Levels. Reddy TE (ed.). *PLoS Genet* 2014;**10**:e1004663.

Barr CL, Misener VL. Decoding the non-coding genome: elucidating genetic risk outside the coding genome. *Genes Brain Behav* 2016;**15**:187–204.

Barski A, Cuddapah S, Cui K *et al.* High-resolution profiling of histone methylations in the human genome. *Cell* 2007;**129**:823–37.

Bedford MT, van Helden PD. Hypomethylation of DNA in pathological conditions of the human prostate. *Cancer Res* 1987;**47**:5274–6.

Bell JT, Pai AA, Pickrell JK *et al.* DNA methylation patterns associate with genetic and gene expression variation in HapMap cell lines. *Genome Biol* 2011;**12**:R10.

Beltran H, Prandi D, Mosquera JM *et al.* Divergent clonal evolution of castration-

resistant neuroendocrine prostate cancer. *Nat Med* 2016;**22**:298–305.

Bennett KL, Mester J, Eng C. Germline epigenetic regulation of KILLIN in Cowden and Cowden-like syndrome. *JAMA* 2010;**304**:2724–31.

Berger SL, Kouzarides T, Shiekhattar R *et al.* An operational definition of epigenetics. *Genes Dev* 2009;**23**:781–3.

Bibikova M, Barnes B, Tsan C *et al.* High density DNA methylation array with single CpG site resolution. *Genomics* 2011;**98**:288–95.

Bojang P Jr, Ramos KS. The promise and failures of epigenetic therapies for cancer treatment. *Cancer Treat Rev* 2014;**40**:153–69.

Bok RA, Small EJ. Bloodborne biomolecular markers in prostate cancer development and progression. *Nat Rev Cancer* 2002;**2**:918–26.

Boström PJ, Bjartell AS, Catto JWF *et al.* Genomic Predictors of Outcome in Prostate Cancer. *Eur Urol* 2015;**68**:1–12.

Brawer MK, Lange PH. PSA in the screening, staging and follow-up of early-stage prostate cancer. *World J Urol* 1989;**7**:7–11.

Breyer JP, Avritt TG, McReynolds KM *et al.* Confirmation of the HOXB13 G84E Germline Mutation in Familial Prostate Cancer. *Cancer Epidemiology Biomarkers & Prevention* 2012;**21**:1348–53.

Brothers C. Huntington's chorea in Victoria and Tasmania. *Journal of the Neurological Sciences* 1964;**1**:405–20.

Browning BL, Browning SR. A Fast, Powerful Method for Detecting Identity by Descent. *The American Journal of Human Genetics* 2011;**88**:173–82.

Burdon KP, McKay JD, Sale MM *et al.* Mutations in a Novel Gene, NHS, Cause the Pleiotropic Effects of Nance-Horan Syndrome, Including Severe Congenital Cataract, Dental Anomalies, and Mental Retardation. *The American Journal of Human Genetics* 2003;**73**:1120–30.

Burgess JR, Nord B, David R *et al.* Phenotype and phenocopy: the relationship between genotype and clinical phenotype in a single large family with multiple endocrine neoplasia type 1 (MEN 1). *Clin Endocrinol (Oxf)* 2000;**53**:205–11.

Carén H, Fransson S, Ejeskär K *et al.* Genetic and epigenetic changes in the common 1p36 deletion in neuroblastoma tumours. *Br J Cancer* 2007;**97**:1416–24.

Carpten J, Nupponen N, Isaacs S *et al.* Germline mutations in the ribonuclease L gene in families showing linkage with HPC1. *Nat Genet* 2002;**30**:181–4.

Carter BS, Beaty TH, Steinberg GD *et al.* Mendelian inheritance of familial

prostate cancer. *Proc Natl Acad Sci USA* 1992;**89**:3367–71.

Cary KC, Cooperberg MR. Biomarkers in prostate cancer surveillance and screening: past, present, and future. *Therapeutic Advances in Urology* 2013;**5**:318–29.

Catalona WJ, Hudson MA, Scardino PT *et al.* Selection of optimal prostate specific antigen cutoffs for early detection of prostate cancer: receiver operating characteristic curves. *J Urol* 1994;**152**:2037–42.

Cazaly E, Charlesworth J, Dickinson JL *et al.* Genetic Determinants of Epigenetic Patterns: Providing insight into disease. *Mol Med* 2015;**21**:400–9.

Cazaly E, Thomson R, Marthick JR *et al.* Comparison of pre-processing methodologies for Illumina 450k methylation array data in familial analyses. *Clin Epigenetics* 2016;**8**:75.

Cedar H, Bergman Y. Programming of DNA methylation patterns. *Annu Rev Biochem* 2012;**81**:97–117.

Center MM, Jemal A, Lortet-Tieulent J *et al.* International Variation in Prostate Cancer Incidence and Mortality Rates. *Eur Urol* 2012;**61**:1079–92.

Chen Y-A, Lemire M, Choufani S *et al.* Discovery of cross-reactive probes and polymorphic CpGs in the Illumina Infinium HumanMethylation450 microarray. *Epigenetics* 2013;**8**:203–9.

Chia N, Wang L, Lu X *et al.* Hypothesis: Environmental regulation of 5-hydroxymethylcytosine by oxidative stress. *Epigenetics* 2014;**6**:853–6.

Chiyomaru T, Yamamura S, Zaman MS *et al.* Genistein Suppresses Prostate Cancer Growth through Inhibition of Oncogenic MicroRNA-151. *PLoS ONE* 2012;**7**, DOI: 10.1371/journal.pone.0043812.

Christensen GB, Bonnie AB, George A. Genome-wide linkage analysis of 1,233 prostate cancer pedigrees from the International Consortium for prostate cancer Genetics using novel sumLINK and sumLOD analyses - Christensen - 2010 - The Prostate - Wiley Online Library. *The …* 2010, DOI: 10.1002/pros.21106/pdf.

Cimmino L, Abdel-Wahab O, Levine RL *et al.* TET Family Proteins and Their Role in Stem Cell Differentiation and Transformation. *Stem Cell* 2011;**9**:193–204.

Cong L, Ran FA, Cox D *et al.* Multiplex Genome Engineering Using CRISPR/Cas Systems. *Science* 2013;**339**:819–23.

Consortium RE, Bilenky M, Heravi-Moussavi A *et al.* Integrative analysis of 111 reference human epigenomes. *Nature* 2015;**518**:317–30.

Cooper DN, Youssoufian H. The CpG dinucleotide and human genetic disease. *Hum Genet* 1988;**78**:151–5.

Cruchaga C, Haller G, Chakraverty S *et al.* Rare variants in APP, PSEN1 and PSEN2 increase risk for AD in late-onset Alzheimer's disease families. Toft M (ed.). *PLoS ONE* 2012;**7**:e31039.

Curran JE, Meikle PJ, Blangero J. New approaches for the discovery of lipid-related genes. *http://dxdoiorg/102217/clp1145* 2011;**6**:495–500.

Dawson MA, Kouzarides T. Cancer Epigenetics: From Mechanism to Therapy. *Cell* 2012;**150**:12–27.

Daxinger L, Whitelaw E. Understanding transgenerational epigenetic inheritance via the gametes in mammals. *Nature Reviews Genetics* 2012;**13**:153–62.

Dayeh TA, Olsson AH, Volkov P *et al.* Identification of CpG-SNPs associated with type 2 diabetes and differential DNA methylation in human pancreatic islets. *Diabetologia* 2013;**56**:1036–46.

Dedeurwaerder S, Defrance M, Calonne E *et al.* Evaluation of the Infinium Methylation 450K technology. *Epigenomics* 2011;**3**:771–84.

Deng GR, Chen AD, Hong J *et al.* Methylation of CpG in a small region of the hMLH1 promoter invariably correlates with the absence of gene expression. *Cancer Res* 1999;**59**:2029–33.

Deng J, Shoemaker R, Xie B *et al.* Targeted bisulfite sequencing reveals changes in DNA methylation associated with nuclear reprogramming. *Nat Biotechnol* 2009;**27**:353–60.

Deorah S, Rao MB, Raman R *et al.* Survival of patients with small cell carcinoma of the prostate during 1973-2003: a population-based study. *BJU Int* 2012;**109**:824–30.

DeWeerdt S. Prognosis: Proportionate response. *Nature* 2015;**528**:S124–5.

Dolinoy DC, Huang D, Jirtle RL. Maternal nutrient supplementation counteracts bisphenol A-induced DNA hypomethylation in early development. *Proc Natl Acad Sci USA* 2007;**104**:13056–61.

Drong AW, Nicholson G, Hedman AK *et al.* The presence of methylation quantitative trait loci indicates a direct genetic influence on the level of DNA methylation in adipose tissue. *PLoS ONE* 2013;**8**:e55923.

Du P, Kibbe WA, Lin SM. lumi: a pipeline for processing Illumina microarray. *Bioinformatics* 2008;**24**:1547–8.

Du P, Zhang X, Huang C-C *et al.* Comparison of Beta-value and M-value methods for quantifying methylation levels by microarray analysis. *BMC Bioinformatics* 2010;**11**:587.

Edwards S, Meitz J, Eles R *et al.* Results of a genome-wide linkage analysis in prostate cancer families ascertained through the ACTANE consortium.

Prostate 2003;**57**:270–9.

Eeles RA, Kote-Jarai Z, Olama Al AA *et al.* Identification of seven new prostate cancer susceptibility loci through a genome-wide association study. *Nature Publishing Group* 2009;**41**:1116–21.

Eeles RA, Olama AAA, Benlloch S *et al.* Identification of 23 new prostate cancer susceptibility loci using the iCOGS custom genotyping array. *Nat Genet* 2013;**45**:385–91.

Eichler EE, Flint J, Gibson G *et al.* Missing heritability and strategies for finding the underlying causes of complex disease. *Nature Reviews Genetics* 2010;**11**:446–50.

Ellinger J, Bastian PJ, Jurgan T *et al.* CpG island hypermethylation at multiple gene sites in diagnosis and prognosis of prostate cancer. *Urology* 2008;**71**:161–7.

ENCODE Project Consortium, Kundaje A, Aldred SF *et al.* An integrated encyclopedia of DNA elements in the human genome. *Nature* 2012;**489**:57–74.

Endo A. A historical perspective on the discovery of statins. *Proc Jpn Acad, Ser B, Phys Biol Sci* 2010;**86**:484–93.

Esteller M. Cancer epigenomics: DNA methylomes and histone-modification maps. *Nature Reviews Genetics* 2007;**8**:286–98.

Ewing CM, Ray AM, Lange EM *et al.* Germline Mutations in HOXB13 and Prostate-Cancer Risk. *N Engl J Med* 2012;**366**:141–9.

Fan S, Li C, Ai R *et al.* Computationally expanding infinium HumanMethylation450 BeadChip array data to reveal distinct DNA methylation patterns of rheumatoid arthritis. *Bioinformatics* 2016;**32**:1773–8.

Feinberg AP, Irizarry RA. Evolution in health and medicine Sackler colloquium: Stochastic epigenetic variation as a driving force of development, evolutionary adaptation, and disease. *PNAS* 2010;**107 Suppl 1**:1757–64.

Feinberg AP, Tycko B. The history of cancer epigenetics. *Nat Rev Cancer* 2004;**4**:143–53.

Feinberg AP. Epigenetic stochasticity, nuclear structure and cancer: the implications for medicine. *J Intern Med* 2014;**276**:5–11.

Ferlay J, Soerjomataram I, Dikshit R *et al.* Cancer incidence and mortality worldwide: Sources, methods and major patterns in GLOBOCAN 2012. *Int J Cancer* 2015;**136**:E359–86.

Fingert JH, Héon E, Liebmann JM *et al.* Analysis of myocilin mutations in 1703

glaucoma patients from five different populations. *Human Molecular Genetics* 1999;**8**:899–905.

FitzGerald LM, Kumar A, Boyle EA *et al.* Germline Missense Variants in the BTNL2 Gene Are Associated with Prostate Cancer Susceptibility. *Cancer Epidemiology Biomarkers & Prevention* 2013;**22**:1520–8.

FitzGerald LM, Patterson B, Thomson R *et al.* Identification of a prostate cancer susceptibility gene on chromosome 5p13q12 associated with risk of both familial and sporadic disease. *Eur J Hum Genet* 2009;**17**:368–77.

Fortin JP, Labbe A, Lemire M *et al.* Functional normalization of 450k methylation array data improves replication in large cancer studies. *bioRxiv* 2014.

Frommer M, McDonald LE, Millar DS *et al.* A genomic sequencing protocol that yields a positive display of 5-methylcytosine residues in individual DNA strands. *Proc Natl Acad Sci USA* 1992;**89**:1827–31.

Furey TS, Sethupathy P. Genetics. Genetics driving epigenetics. *Science* 2013;**342**:705–6.

Gagnon-Bartsch JA, Jacob L, Speed TP. Removing Unwanted Variation from High Dimensional Data with Negative Controls. 2012:1–104.

Gamazon ER, Badner JA, Cheng L *et al.* Enrichment of cis-regulatory gene expression SNPs and methylation quantitative trait loci among bipolar disorder susceptibility variants. *Mol Psychiatry* 2012;**18**:340–6.

Gaunt TR, Shihab HA, Hemani G *et al.* Systematic identification of genetic influences on methylation across the human life course. *Genome Biol* 2016;**17**:1.

Gerrish A, Russo G, Richards A *et al.* The role of variation at AβPP, PSEN1, PSEN2, and MAPT in late onset Alzheimer's disease. *J Alzheimers Dis* 2012;**28**:377–87.

Gertz J, Varley KE, Reddy TE *et al.* Analysis of DNA Methylation in a Three-Generation Family Reveals Widespread Genetic Influence on Epigenetic Regulation. Bickmore WA (ed.). *PLoS Genet* 2011;**7**:e1002228.

Ghadirian P, Howe GR, Hislop TG *et al.* Family history of prostate cancer: A multi-center case-control study in Canada. *Int J Cancer* 1997;**70**:679–81.

Gibbs JR, van der Brug MP, Hernandez DG *et al.* Abundant quantitative trait loci exist for DNA methylation and gene expression in human brain. *PLoS Genet* 2010;**6**:e1000952.

Gimelbrant A, Hutchinson JN, Thompson BR *et al.* Widespread Monoallelic Expression on Human Autosomes. *Science* 2007;**318**:1136–40.

Graff JR, Herman JG, Myöhänen S *et al.* Mapping patterns of CpG island

methylation in normal and neoplastic cells implicates both upstream and downstream regions in de novo methylation. *J Biol Chem* 1997;**272**:22322–9.

Grigore AD, Ben-Jacob E, Farach-Carson MC. Prostate cancer and neuroendocrine differentiation: more neuronal, less endocrine? *Front Oncol* 2015;**5**:37.

Gronberg H, Damber L, Damber JE. Familial prostate cancer in sweden: A nationwide register cohort study. *Cancer* 1996;**77**:138–43.

Guerrero-Bosagna C, Skinner MK. Environmentally induced epigenetic transgenerational inheritance of phenotype and disease. *MOLECULAR AND CELLULAR ENDOCRINOLOGY* 2011;**354**:1–6.

Hackett JA, Sengupta R, Zylicz JJ *et al.* Germline DNA Demethylation Dynamics and Imprint Erasure Through 5-Hydroxymethylcytosine. *Science* 2013;**339**:448–52.

Haffner MC, Chaux A, Meeker AK *et al.* Global 5-hydroxymethylcytosine content is significantly reduced in tissue stem/progenitor cell compartments and in human cancers. *Oncotarget* 2011;**2**:627–37.

Hansen KD, Timp W, Bravo HC *et al.* Increased methylation variation in epigenetic domains across cancer types. *Nat Genet* 2011;**43**:768–75.

Harper KN, Peters BA, Gamble MV. Batch effects and pathway analysis: two potential perils in cancer studies involving DNA methylation array analysis. *Cancer Epidemiol Biomarkers Prev* 2013;**22**:1052–60.

Hebestreit K, Klein HU. BiSeq: A package for analyzing targeted bisulfite sequencing data. 2013.

Hellman A. Extensive sequence-influenced DNA methylation polymorphism in the human genome. *Epigenetics Chromatin* 2010;**3**:11.

Hesson LB, Hitchins MP, Ward RL. Epimutations and cancer predisposition: importance and mechanisms. *Current Opinion in Genetics & Development* 2010;**20**:290–8.

Heyn H, Moran S, Hernando-Herraez I *et al.* DNA methylation contributes to natural human variation. *Genome Research* 2013;**23**:1363–72.

Heyn H, Sayols S, Moutinho C *et al.* Linkage of DNA Methylation Quantitative Trait Loci to Human Cancer Risk. *CellReports* 2014;**7**:1–8.

Heyn H. Quantitative Trait Loci Identify Functional Noncoding Variation in Cancer. Greally JM (ed.). *PLoS Genet* 2016;**12**:e1005826.

Hindorff LA, Sethupathy P, Junkins HA *et al.* Potential etiologic and functional implications of genome-wide association loci for human diseases and traits. *PNAS* 2009;**106**:9362–7.

Hitchins MP, Lynch HT. Dawning of the epigenetic era in hereditary cancer. *Clin Genet* 2014;**85**:413–6.

Hitchins MP, Rapkins RW, Kwok C-T *et al.* Dominantly inherited constitutional epigenetic silencing of MLH1 in a cancer-affected family is linked to a single nucleotide variant within the 5'UTR. *Cancer Cell* 2011;**20**:200–13.

Hjelmborg JB, Scheike T, Holst K *et al.* The Heritability of Prostate Cancer in the Nordic Twin Study of Cancer. *Cancer Epidemiology Biomarkers & Prevention* 2014;**23**:2303–10.

Hodson R. Prostate cancer: 4 big questions. *Nature* 2015;**528**:S137–7.

Hopkins TG, Burns PA, Routledge MN. DNA Methylation of GSTP1 as Biomarker in Diagnosis of Prostate Cancer. *Urology* 2007;**69**:11–6.

Ianni M, Porcellini E, Carbone I *et al.* Genetic factors regulating inflammation and DNA methylation associated with prostate cancer. *Prostate Cancer and Prostatic Disease* 2012;**16**:56–61.

Illingworth RS, Gruenewald-Schneider U, Webb S *et al.* Orphan CpG Islands Identify Numerous Conserved Promoters in the Mammalian Genome. *PLoS Genet* 2010;**6**, DOI: 10.1371/journal.pgen.1001134.

Illumina. HumanMethylation450 BeadChip. *Nature* 2012;**448**:1–4.

Irizarry RA, Ladd-Acosta C, Wen B *et al.* The human colon cancer methylome shows similar hypo- and hypermethylation at conserved tissue-specific CpG island shores. *Nature Publishing Group* 2009;**41**:178–86.

Ito S, D'Alessio AC, Taranova OV *et al.* Role of Tet proteins in 5mC to 5hmC conversion, ES-cell self-renewal and inner cell mass specification. *Nature* 2010;**466**:1129–33.

Jacobs EJ, Chanock SJ, Fuchs CS *et al.* Family history of cancer and risk of pancreatic cancer: a pooled analysis from the Pancreatic Cancer Cohort Consortium (PanScan). *International Journal of Cancer* 2010;**127**:1421–8.

Jin G, Lu L, Cooney KA *et al.* Validation of prostate cancer risk-related loci identified from genome-wide association studies using family-based association analysis: evidence from the International Consortium for Prostate Cancer Genetics (ICPCG). *Hum Genet* 2012;**131**:1095–103.

Jin G, Sun J, Isaacs SD *et al.* Human polymorphisms at long non-coding RNAs (lncRNAs) and association with prostate cancer risk. *Carcinogenesis* 2011;**32**:1655–9.

Jjingo D, Conley AB, Yi SV *et al.* On the presence and role of human gene-body DNA methylation. *Oncotarget* 2012;**3**:462–74.

Johnson AD, Handsaker RE, Pulit SL *et al.* SNAP: a web-based tool for

identification and annotation of proxy SNPs using HapMap. *Bioinformatics* 2008;**24**:2938–9.

Jones PA, Baylin SB. The Epigenomics of Cancer. *Cell* 2007;**128**:683–92.

Jones PA. The DNA methylation paradox. *Trends Genet* 1999;**15**:34–7.

Jones PA. Functions of DNA methylation: islands, start sites, gene bodies and beyond. *Nature Reviews Genetics* 2012;**13**:484–92.

Jones PA. At the tipping point for epigenetic therapies in cancer. *J Clin Invest* 2014;**124**:14–6.

Jung M, Pfeifer GP. Aging and DNA methylation. *BMC Biology 2015 13:1* 2015;**13**:1.

Kang GH, Lee S, Lee HJ *et al.* Aberrant CpG island hypermethylation of multiple genes in prostate cancer and prostatic intraepithelial neoplasia. *J Pathol* 2004;**202**:233–40.

Kasinski AL, Slack FJ. MicroRNAs en route to the clinic: progress in validating and targeting microRNAs for cancer therapy. *Nat Rev Cancer* 2011;**11**:849–64.

Kasowski M, Grubert F, Heffelfinger C *et al.* Variation in Transcription Factor Binding Among Humans. *Science* 2010;**328**:232–5.

Kasowski M, Kyriazopoulou-Panagiotopoulou S, Grubert F *et al.* Extensive Variation in Chromatin States Across Humans. *Science* 2013;**342**:750–2.

Keetch DW, Rice JP, Suarez BK *et al.* Familial Aspects of Prostate Cancer: A Case Control Study. *J Urol* 1995;**154**:2100–2.

Kent WJ, Sugnet CW, Furey TS *et al.* The human genome browser at UCSC. *Genome Research* 2002;**12**:996–1006.

Kicinski M, Vangronsveld J, Nawrot TS. An Epidemiological Reappraisal of the Familial Aggregation of Prostate Cancer: A Meta-Analysis. Little J (ed.). *PLoS ONE* 2011;**6**, DOI: 10.1371/journal.pone.0027130.

Kilpinen H, Waszak SM, Gschwind AR *et al.* Coordinated Effects of Sequence Variation on DNA Binding, Chromatin Structure, and Transcription. *Science* 2013;**342**:744–7.

Kirchhoff T, Kauff ND, Mitra N *et al.* BRCA Mutations and Risk of Prostate Cancer in Ashkenazi Jews. *Clin Cancer Res* 2004;**10**:2918–21.

Kote-Jarai Z, Mikropoulos C, Leongamornlert DA *et al.* Prevalence of the HOXB13 G84E germline mutation in British men and correlation with prostate cancer risk, tumour characteristics and clinical outcomes. *Ann Oncol* 2015;**26**:756–61.

Koul HK, Kumar B, Koul S *et al.* The role of inflammation and infection in prostate cancer: Importance in prevention, diagnosis and treatment. *Drugs Today* 2010;**46**:929–43.

Kriaucionis S, Heintz N. The nuclear DNA base 5-hydroxymethylcytosine is present in Purkinje neurons and the brain. *Science* 2009;**324**:929–30.

Krueger F, Andrews SR. Bismark: a flexible aligner and methylation caller for Bisulfite-Seq applications. *Bioinformatics* 2011;**27**:1571–2.

Kulis M, Heath S, Bibikova M *et al.* Epigenomic analysis detects widespread gene-body DNA hypomethylation in chronic lymphocytic leukemia. *Nature Publishing Group* 2012;**44**:1236–42.

Kulkarni H, Kos MZ, Neary J *et al.* Novel epigenetic determinants of type 2 diabetes in Mexican-American families. *Human Molecular Genetics* 2015;**24**:5330–44.

Kumar V, Westra H-J, Karjalainen J *et al.* Human disease-associated genetic variation impacts large intergenic non-coding RNA expression. *PLoS Genet* 2013;**9**:e1003201.

Lange EM, Gillanders EM, Davis CC *et al.* Genome-wide scan for prostate cancer susceptibility genes using families from the University of Michigan prostate cancer genetics project finds evidence for linkage on chromosome 17 near BRCA1. *Prostate* 2003;**57**:326–34.

Leek JT, Johnson WE, Parker HS *et al.* The sva package for removing batch effects and other unwanted variation in high-throughput experiments. *Bioinformatics* 2012;**28**:882–3.

Lemire M, Zaidi SHE, Ban M *et al.* Long-range epigenetic regulation is conferred by genetic variation located at thousands of independent loci. *Nat Commun* 2015;**6**, DOI: 10.1038/ncomms7326.

Leyten GHJM, Hessels D, Jannink SA *et al.* Prospective Multicentre Evaluation of PCA3 and TMPRSS2-ERG Gene Fusions as Diagnostic and Prognostic Urinary Biomarkers for Prostate Cancer. *Eur Urol* 2014;**65**:534–42.

Li L-C, Dahiya R. MethPrimer: designing primers for methylation PCRs. *Bioinformatics* 2002;**18**:1427–31.

Lichtensztajn DY, Gomez SL, Sieh W *et al.* Prostate Cancer Risk Profiles of Asian-American Men: Disentangling the Effects of Immigration Status and Race/Ethnicity. *J Urol* 2014;**191**:952–6.

Liebers R, Rassoulzadegan M, Lyko F. Epigenetic Regulation by Heritable RNA. *PLoS Genet* 2014;**10**:e1004296.

Liu Z, Yang X, Li Z *et al.* CASZ1, a candidate tumor-suppressor gene, suppresses neuroblastoma tumor growth through reprogramming gene expression. *Cell*

Death Differ 2011;**18**:1174–83.

Liu ZH, Yang XZ, Tan F *et al.* Molecular cloning and characterization of human Castor, a novel human gene upregulated during cell differentiation. *Biochem Biophys Res Commun* 2006;**344**:834–44.

Loeb S, Peskoe SB, Joshu CE *et al.* Do Environmental Factors Modify the Genetic Risk of Prostate Cancer? *Cancer Epidemiology Biomarkers & Prevention* 2015;**24**:213–20.

Luijk R, Goeman JJ, Slagboom EP *et al.* An alternative approach to multiple testing for methylation QTL mapping reduces the proportion of falsely identified CpGs. *Bioinformatics* 2015;**31**:340–5.

Maksimovic J, Gordon L, Oshlack A. SWAN: Subset-quantile Within Array Normalization for Illumina Infinium HumanMethylation450 BeadChips. *Genome Biol* 2012;**13**:R44.

Maldonado L, Brait M, Loyo M *et al.* GSTP1 Promoter Methylation is Associated with Recurrence in Early Stage Prostate Cancer. *J Urol* 2014;**192**:1542–8.

Mali P, Yang L, Esvelt KM *et al.* RNA-guided human genome engineering via Cas9. *…* 2013, DOI: 10.1126/science.1231143.

Manolio TA, Collins FS, Cox NJ *et al.* Finding the missing heritability of complex diseases. *Nature* 2009;**461**:747–53.

Marabita F, Almgren M, Lindholm ME *et al.* An evaluation of analysis pipelines for DNA methylation profiling using the Illumina HumanMethylation450 BeadChip platform. *Epigenetics* 2013;**8**:333–46.

Marchani EE, Chapman NH, Cheung CYK *et al.* Identification of Rare Variants from Exome Sequence in a Large Pedigree with Autism. *Hum Hered* 2012;**74**:153–64.

Marjoram P, Zubair A, Nuzhdin SV. Post-GWAS: where next|[quest]| More samples, more SNPs or more biology|[quest]|. *Heredity* 2014;**112**:79–88.

Martin M. Cutadapt removes adapter sequences from high-throughput sequencing reads. *EMBnetjournal* 2011;**17**:pp.10–2.

Maunakea AK, Chepelev I, Cui K *et al.* Intragenic DNA methylation modulates alternative splicing by recruiting MeCP2 to promote exon recognition. *Cell Res* 2013;**23**:1256–69.

Maunakea AK, Nagarajan RP, Bilenky M *et al.* Conserved role of intragenic DNA methylation in regulating alternative promoters. *Nature* 2010;**466**:253–U131.

McKay JA, Groom A, Potter C *et al.* Genetic and Non-Genetic Influences during Pregnancy on Infant Global and Site Specific DNA Methylation: Role for

Folate Gene Variants and Vitamin B 12. Rishi A (ed.). *PLoS ONE* 2012;**7**:e33290.

McVicker G, van de Geijn B, Degner JF *et al.* Identification of Genetic Variants That Affect Histone Modifications in Human Cells. *Science* 2013;**342**:747–9.

Moore LE, Nickerson ML, Brennan P *et al.* Von Hippel-Lindau (VHL) inactivation in sporadic clear cell renal cancer: associations with germline VHL polymorphisms and etiologic risk factors. Maher ER (ed.). *PLoS Genet* 2011;**7**:e1002312.

Morris TJ, Beck S. Analysis pipelines and packages for Infinium HumanMethylation450 BeadChip (450k) data. *Methods* 2015;**72**:3–8.

Morris TJ, Butcher LM, Feber A *et al.* ChAMP: 450k Chip Analysis Methylation Pipeline. *Bioinformatics* 2014;**30**:428–30.

Ogino S, Hazra A, Tranah GJ *et al.* MGMT germline polymorphism is associated with somatic MGMT promoter methylation and gene silencing in colorectal cancer. *Carcinogenesis* 2007;**28**:1985–90.

Olama Al AA, Kote-Jarai Z, Berndt SI *et al.* A meta-analysis of 87,040 individuals identifies 23 new susceptibility loci for prostate cancer. *Nat Genet* 2014;**46**:1103–9.

Olsson AH, Volkov P, Bacos K *et al.* Genome-Wide Associations between Genetic and Epigenetic Variation Influence mRNA Expression and Insulin Secretion in Human Pancreatic Islets. *PLoS Genet* 2014;**10**:e1004735.

Oyer JA, Chu A, Brar S *et al.* Aberrant epigenetic silencing is triggered by a transient reduction in gene expression. *PLoS ONE* 2009;**4**:e4832.

Panagiotou OA, Ioannidis JPA, Project G-WS. What should the genome-wide significance threshold be? Empirical replication of borderline genetic associations. *Int J Epidemiol* 2012;**41**:273–86.

Park H, Kim H-J, Lee S *et al.* A family-based association study after genome-wide linkage analysis identified two genetic loci for renal function in a Mongolian population. *Kidney Int* 2013;**83**:285–92.

Patrikidou A, Loriot Y, Eymard J-C *et al.* Who dies from prostate cancer&quest. *Prostate Cancer and Prostatic Disease* 2014;**17**:348–52.

Pattaro C, Saint-Pierre A. Family-based studies to the rescue of genome-wide association studies in renal function. *Kidney Int* 2013;**83**:196–8.

Patterson K, Molloy L, Qu W *et al.* DNA methylation: bisulphite modification and analysis. *J Vis Exp* 2011, DOI: 10.3791/3170.

Pembrey M, Saffery R, Bygren LO *et al.* Human transgenerational responses to early-life experience: potential impact on development, health and

biomedical research. *J Med Genet* 2014;**51**:563–72.

Penson DF, McLerran D, Feng Z *et al.* 5-year urinary and sexual outcomes after radical prostatectomy: results from the prostate cancer outcomes study. *J Urol* 2005;**173**:1701–5.

Pidsley R, Y Wong CC, Volta M *et al.* A data-driven approach to preprocessing Illumina 450K methylation array data. *BMC Genomics* 2013;**14**:293.

Prensner JR, Rubin MA, Wei JT *et al.* Beyond PSA: the next generation of prostate cancer biomarkers. *Science Translational Medicine* 2012;**4**:127rv3–127rv3.

Purcell S, Neale B, Todd-Brown K *et al.* PLINK: A Tool Set for Whole-Genome Association and Population-Based Linkage Analyses. *The American Journal of Human Genetics* 2007;**81**:559–75.

Ramalho-Carvalho J, Fromm B, Henrique R *et al.* Deciphering the function of non-coding RNAs in prostate cancer. *Cancer Metastasis Rev* 2016;**35**:235–62.

Rhee H, Vela I, Chung E. Metabolic Syndrome and Prostate Cancer: a Review of Complex Interplay Amongst Various Endocrine Factors in the Pathophysiology and Progression of Prostate Cancer. *Horm Cancer* 2016;**7**:75–83.

Richards EJ, Elgin SCR. Epigenetic Codes for Heterochromatin Formation and Silencing. *Cell* 2002;**108**:489–500.

Richards EJ. Inherited epigenetic variation--revisiting soft inheritance. *Nature Reviews Genetics* 2006;**7**:395–401.

Richman EL, Kenfield SA, Stampfer MJ *et al.* Egg, red meat, and poultry intake and risk of lethal prostate cancer in the prostate-specific antigen-era: incidence and survival. *Cancer Prevention Research* 2011;**4**:2110–21.

Rideout WM, Coetzee GA, Olumi AF *et al.* 5-Methylcytosine as an endogenous mutagen in the human LDL receptor and p53 genes. *Science* 1990;**249**:1288–90.

Roobol M. Perspective: Enforce the clinical guidelines. *Nature* 2015;**528**:S123–3.

Rushton MD, Reynard LN, Young DA *et al.* Methylation quantitative trait locus analysis of osteoarthritis links epigenetics with genetic risk. *Human Molecular Genetics* 2015;**24**:7432–44.

Saad M, Wijsman EM. Power of Family-Based Association Designs to Detect Rare Variants in Large Pedigrees Using Imputed Genotypes. *Genet Epidemiol* 2014;**38**:1–9.

Saini S. PSA and beyond: alternative prostate cancer biomarkers. *Cell Oncol (Dordr)* 2016;**39**:1–10.

Sandoval J, Heyn H, Moran S *et al.* Validation of a DNA methylation microarray for 450,000 CpG sites in the human genome. *Epigenetics* 2011;**6**:692–702.

Sandoval J, Heyn H, Moran S *et al.* Validation of a DNA methylation microarray for 450,000 CpG sites in the human genome. *Epigenetics* 2014;**6**:692–702.

Sasaki H, Matsui Y. Epigenetic events in mammalian germ-cell development: reprogramming and beyond. *Nature Reviews Genetics* 2008;**9**:129–40.

Savage N. Metastasis: Resistance fighters. *Nature* 2015;**528**:S128–9.

Seisenberger S, Andrews S, Krueger F *et al.* The dynamics of genome-wide DNA methylation reprogramming in mouse primordial germ cells. *Mol Cell* 2012;**48**:849–62.

Shannon J, Phoutrides E, Palma A *et al.* Folate Intake and Prostate Cancer Risk: A Case-Control Study. *Nutrition and Cancer-an International Journal* 2009;**61**:617–28.

Sharma S, Kelly TK, Jones PA. Epigenetics in cancer. *Carcinogenesis* 2010;**31**:27–36.

Shen H, Fridley BL, Song H *et al.* Epigenetic analysis leads to identification of HNF1B as a subtype-specific susceptibility gene for ovarian cancer. *Nat Commun* 2013;**4**, DOI: 10.1038/ncomms2629.

Shoemaker R, Deng J, Wang W *et al.* Allele-specific methylation is prevalent and is contributed by CpG-SNPs in the human genome. *Genome Research* 2010;**20**:883–9.

Shukla S, Kavak E, Gregory M *et al.* CTCF-promoted RNA polymerase II pausing links DNA methylation to splicing. *Nature* 2011;**479**:74–9.

Skinner MK. Environmental epigenetic transgenerational inheritance and somatic epigenetic mitotic stability. *Epigenetics* 2011;**6**:838–42.

Smith AK, Kilaru V, Kocak M *et al.* Methylation quantitative trait loci (meQTLs) are consistently detected across ancestry, developmental stage, and tissue type. *BMC Genomics* 2014;**15**:145.

Sohn E. Screening: Diagnostic dilemma. *Nature* 2015;**528**:S120–2.

Song JZ, Stirzaker C, Harrison J *et al.* Hypermethylation trigger of the glutathione-S-transferase gene (GSTP1) in prostate cancer cells. *Oncogene* 2002;**21**:1048–61.

Sproul D, Meehan RR. Genomic insights into cancer-associated aberrant CpG island hypermethylation. *Briefings in Functional Genomics* 2013;**12**:174–90.

Stirzaker C, Taberlay PC, Statham AL *et al.* Mining cancer methylomes: prospects and challenges. *Trends Genet* 2014;**30**:75–84.

Sun Z, Chai HS, Wu Y *et al.* Batch effect correction for genome-wide methylation data with Illumina Infinium platform. *BMC Med Genomics* 2011;**4**:84.

Suzuki MM, Bird A. DNA methylation landscapes: provocative insights from epigenomics. *Nature Reviews Genetics* 2008;**9**:465–76.

Taqi MM, Bazov I, Watanabe H *et al.* Prodynorphin CpG-SNPs associated with alcohol dependence: elevated methylation in the brain of human alcoholics. *Addict Biol* 2011;**16**:499–509.

Tarapore P, Ying J, Ouyang B *et al.* Exposure to bisphenol A correlates with early-onset prostate cancer and promotes centrosome amplification and anchorage-independent growth in vitro. Kyprianou N (ed.). *PLoS ONE* 2014;**9**:e90332.

Teerlink CC, Leongamornlert D, Dadaev T *et al.* Genome-wide association of familial prostate cancer cases identifies evidence for a rare segregating haplotype at 8q24.21. *Hum Genet* 2016:1–16.

Teerlink CC, Thibodeau SN, McDonnell SK *et al.* Association analysis of 9,560 prostate cancer cases from the International Consortium of Prostate Cancer Genetics confirms the role of reported prostate cancer associated SNPs for familial disease. *Hum Genet* 2014;**133**:347–56.

Teschendorff AE, Jones A, Fiegl H *et al.* Epigenetic variability in cells of normal cytology is associated with the risk of future morphological transformation. *Genome Med* 2012;**4**:24.

Teschendorff AE, Marabita F, Lechner M *et al.* A beta-mixture quantile normalization method for correcting probe design bias in Illumina Infinium 450 k DNA methylation data. *Bioinformatics* 2013;**29**:189–96.

Teschendorff AE, Widschwendter M. Differential variability improves the identification of cancer risk markers in DNA methylation studies profiling precursor cancer lesions. *Bioinformatics* 2012;**28**:1487–94.

Thomas DC. Some Surprising Twists on the Road to Discovering the Contribution of Rare Variants to Complex Diseases. *Hum Hered* 2012;**74**:113–7.

Timp W, Bravo HC, McDonald OG *et al.* Large hypomethylated blocks as a universal defining epigenetic alteration in human solid tumors. *Genome Med* 2014;**6**, DOI: 10.1186/s13073-014-0061-y.

Tomso DJ, Bell DA. Sequence Context at Human Single Nucleotide Polymorphisms: Overrepresentation of CpG Dinucleotide at Polymorphic Sites and Suppression of Variation in CpG Islands. *Journal of Molecular Biology* 2003;**327**:303–8.

Touleimat N, Tost J. Complete pipeline for Infinium(®) Human Methylation 450K BeadChip data processing using subset quantile normalization for accurate DNA methylation estimation. *Epigenomics* 2012;**4**:325–41.

Triche TJ, Weisenberger DJ, Van Den Berg D *et al.* Low-level processing of Illumina Infinium DNA Methylation BeadArrays. *Nucleic Acids Res* 2013;**41**:e90–0.

Varley KE, Gertz J, Bowling KM *et al.* Dynamic DNA methylation across diverse human cell lines and tissues. *Genome Research* 2013;**23**:555–67.

Veenendaal MVE, Painter RC, de Rooij SR *et al.* Transgenerational effects of prenatal exposure to the 1944-45 Dutch famine. *BJOG* 2013;**120**:548–53.

Waddington CH. The Epigenotype. *Int J Epidemiol* 2012;**41**:10–3.

Wang C, Liu Z, Woo C-W *et al.* EZH2 Mediates Epigenetic Silencing of Neuroblastoma Suppressor Genes CASZ1, CLU, RUNX3, and NGFR. *Cancer Res* 2012;**72**:315–24.

Wang Y, Leung FCC. An evaluation of new criteria for CpG islands in the human genome as gene markers. *Bioinformatics* 2004;**20**:1170–7.

Ward RL, Dobbins T, Lindor NM *et al.* Identification of constitutional MLH1 epimutations and promoter variants in colorectal cancer patients from the Colon Cancer Family Registry. *Genet Med* 2013;**15**:25–35.

Whitington T, Gao P, Song W *et al.* Gene regulatory mechanisms underpinning prostate cancer susceptibility. *Nature Publishing Group* 2016, DOI: 10.1038/ng.3523.

Wiewrodt D, Nagel G, Dreimüller N *et al.* MGMT in primary and recurrent human glioblastomas after radiation and chemotherapy and comparison with p53 status and clinical outcome. *Int J Cancer* 2007;**122**:1391–9.

Wijsman EM. The role of large pedigrees in an era of high-throughput sequencing. *Hum Genet* 2012;**131**:1555–63.

Williams JT, Blangero J. Power of variance component linkage analysis to detect quantitative trait loci. *Ann Hum Genet* 1999;**63**:545–63.

Woodson K, O'Reilly KJ, Hanson JC *et al.* The Usefulness of the Detection of GSTP1 Methylation in Urine as a Biomarker in the Diagnosis of Prostate Cancer. *J Urol* 2008;**179**:508–12.

Wu K, Spiegelman D, Hou T *et al.* Associations between unprocessed red and processed meat, poultry, seafood and egg intake and the risk of prostate cancer: A pooled analysis of 15 prospective cohort studies. *International Journal of Cancer* 2016;**138**:2368–82.

Xu JF, Zheng SL, Komiya A *et al.* Germline mutations and sequence variants of the macrophage scavenger receptor 1 gene are associated with prostate cancer risk. *Nat Genet* 2002;**32**:321–5.

Yan PS, Shi HD, Rahmatpanah F *et al.* Differential distribution of DNA

methylation within the RASSF1A CpG island in breast cancer. *Cancer Res* 2003;**63**:6178–86.

Yang X, Han H, De Carvalho DD *et al.* Gene Body Methylation Can Alter Gene Expression and Is a Therapeutic Target in Cancer. *Cancer Cell* 2014;**26**:577–90.

Yegnasubramanian S, Haffner MC, Zhang Y *et al.* DNA hypomethylation arises later in prostate cancer progression than CpG island hypermethylation and contributes to metastatic tumor heterogeneity. *Cancer Res* 2008;**68**:8954–67.

You JS, Jones PA. Cancer Genetics and Epigenetics: Two Sides of the Same Coin? *Cancer Cell* 2012;**22**:9–20.

Zhang D, Cheng L, Badner JA *et al.* Genetic Control of Individual Differences in Gene-Specific Methylation in Human Brain. *The American Journal of Human Genetics* 2010;**86**:411–9.

Zhi D, Aslibekyan S, Irvin MR *et al.* SNPs located at CpG sites modulate genome-epigenome interaction. *Epigenetics* 2013;**8**:802–6.

Zhou D, Li Z, Yu D *et al.* Polymorphisms involving gain or loss of CpG sites are significantly enriched in trait-associated SNPs. *Oncotarget* 2015;**6**:39995–40004.

Ziegler A, Sun YV. Study designs and methods post genome-wide association studies. *Hum Genet* 2012;**131**:1525–31.

MolecularMedicine

Genetic Determinants of Epigenetic Patterns: Providing Insight into Disease

Emma Cazaly,[1] Jac Charlesworth,[1] Joanne L Dickinson,[1] and Adele F Holloway[2]

[1]Menzies Institute for Medical Research, University of Tasmania, Hobart, Tasmania, Australia; and the [2]School of Medicine, University of Tasmania, Hobart, Tasmania, Australia

The field of epigenetics and our understanding of the mechanisms that regulate the establishment, maintenance and heritability of epigenetic patterns continue to grow at a remarkable rate. This information is providing increased understanding of the role of epigenetic changes in disease, insight into the underlying causes of these epigenetic changes and revealing new avenues for therapeutic intervention. Epigenetic modifiers are increasingly being pursued as therapeutic targets in a range of diseases, with a number of agents targeting epigenetic modifications already proving effective in diseases such as cancer. Although it is well established that DNA mutations and aberrant expression of epigenetic modifiers play a key role in disease, attention is now turning to the interplay between genetic and epigenetic factors in complex disease etiology. The role of genetic variability in determining epigenetic profiles, which can then be modified by environmental and stochastic factors, is becoming more apparent. Understanding the interplay between genetic and epigenetic factors is likely to aid in identifying individuals most likely to benefit from epigenetic therapies. This goal is coming closer to realization because of continual advances in laboratory and statistical tools enabling improvements in the integration of genomic, epigenomic and phenotypic data.

Online address: http://www.molmed.org
doi: 10.2119/molmed.2015.00001

WHY EPIGENETICS AND WHY NOW?

The genomics era brought with it dramatic advances in our understanding of the molecular basis of disease. High-density genome mapping strategies have proven particularly successful for the identification of genes underlying mendelian disorders, such as hemochromatosis, cystic fibrosis and muscular dystrophy (1). The advent of genome-wide association studies (GWAS) was heralded with the promise of providing a comprehensive map of genetic susceptibility to complex disease. While uncovering thousands of variants associated with disease risk, their promise is yet to be fully realized, with a persistent gap emerging between the fraction of disease accounted for by genetic variation and the heritability estimates for many traits (2). Several explanations have been proposed for this unexplained genetic component to disease susceptibility, including the impact of large deletions, inversions or copy number variants, complex gene–gene and gene–environment interactions, overestimated heritability, poor modeling and statistical application and common variants masking rare variants or driving synthetic association (2). Notably, deleterious variants occurring in coding regions account for the minority of disease-associated single nucleotide polymorphisms (SNPs), with estimates that over 90% of variants identified in GWAS are located in noncoding regions of the genome (3). At least some of these SNPs affect gene regulatory mechanisms, modifying gene expression by altering transcription factor binding and directing altered epigenetic profiles (4).

Studies involving genetically identical individuals such as monozygotic (MZ) twins have been invaluable in investigating the role of genetics and environment in complex disease. While providing insight into the genetic basis of disease, these studies have also strongly implicated a nongenetic contribution to many diseases. For example, MZ twins have much less than 100% concordance rates for common diseases such as Alzheimer's disease and certain cancers (5). The effect of the environment and random factors on the epigenome poses a possible explanation for this discordance, since while MZ twin epigenetic profiles show a high level of heritability early in development, they diverge with age and differing lifestyles and epigenetic marks differ according to disease state (5).

Address correspondence to *Adele F Holloway, School of Medicine, University of Tasmania, Private Bag 34, Hobart Tasmania 7000, Australia. Phone: +61-(0)3-6226-2670; Fax: +61-(0)3-6226-7704; E-mail: a.f.holloway@utas.edu.au.*
Submitted January 6, 2015; Accepted for publication March 26, 2015; Published Online (www.molmed.org) March 26, 2015.

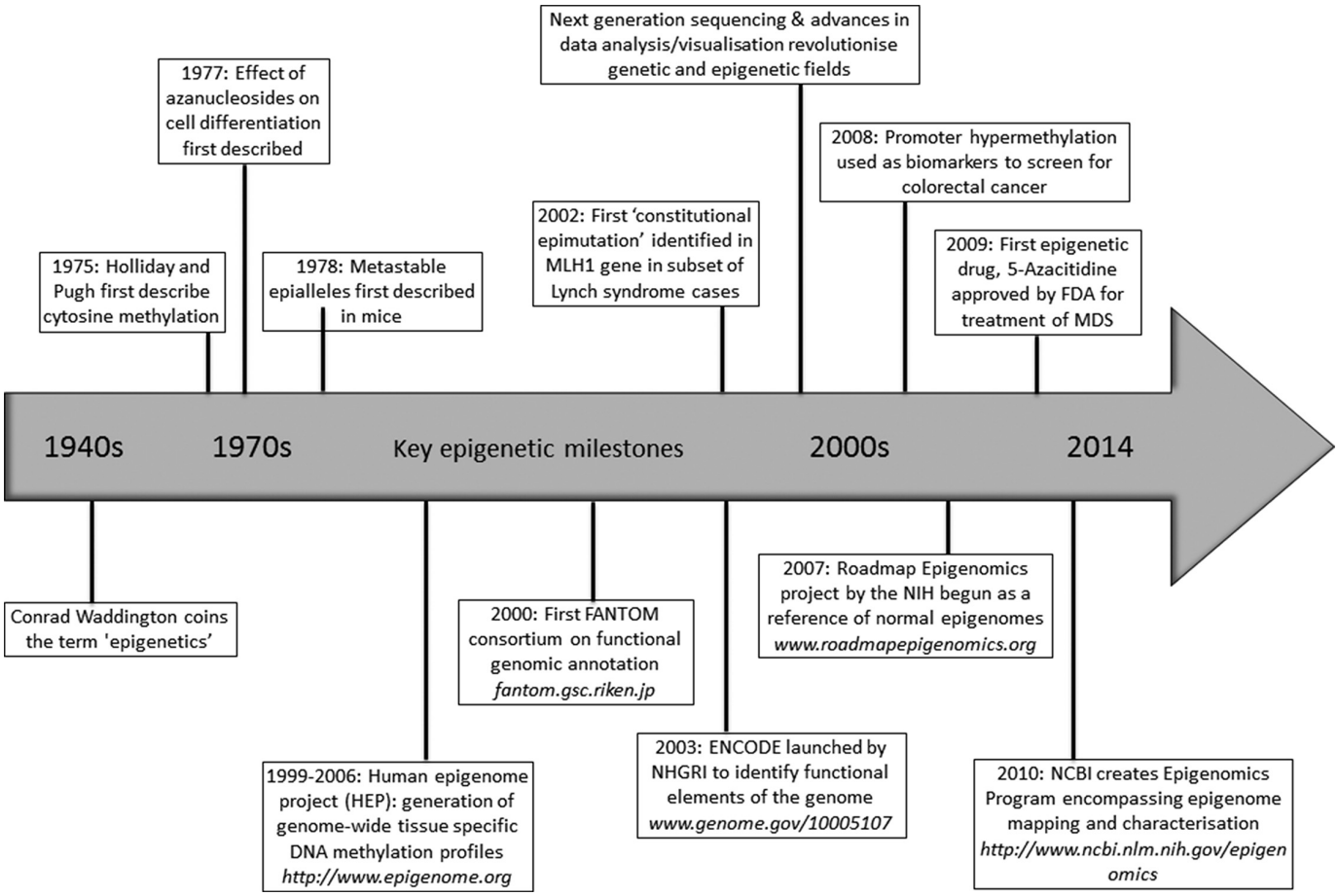


Figure 1. Timeline of key advances in the epigenetics field. Key milestones in the epigenetics field are shown.

Our understanding of the factors affecting the epigenome is increasing at a rapid rate. The role of the underlying genetic sequence in determining epigenetic profiles and the mechanisms by which environmental and stochastic factors then modify these epigenetic patterns are becoming clearer. Our increased understanding of these mechanisms and their role in disease processes is being driven by rapid advances in laboratory and statistical tools and the creation of extensive public databases. This result makes it possible to integrate genomic, epigenomic and phenotypic data with a greater level of detail and scale than ever before (6) (see Figure 1 for a timeline of epigenetic milestones, including the advent of online annotation databases). This information is providing a valuable resource for the investigation of complex disease, revealing new opportunities for disease prevention in at-risk individuals, identifying new therapeutic targets and providing the prospect of increased sensitivity and specificity of disease diagnosis (7).

A BRIEF HISTORY OF EPIGENETICS

Conrad Waddington first coined the term "epigenetics" in the early 1940s to integrate the existence of two related phenomena: that genetically identical cells possess the capacity to differentiate into tissue-specific structures with correlated functions and that gene–environment interactions can affect phenotypes (reprinted in Waddington [8]). The term "epigenetics" has since come to refer to the environment surrounding the DNA, with a current working definition characterizing an epigenetic trait as "a stably heritable phenotype resulting from changes in a chromosome without alterations in the DNA sequence" (9). This term is most often used in reference to the inheritance of traits to a daughter cell during mitosis, but there is evidence, although still controversial, of germ line transmission of epigenetic traits between generations (transgenerational inheritance) (10,11).

Eukaryotic DNA is assembled into chromatin; repeating units of nucleosomes consisting of 147 base pairs of DNA wrapped around an octamer of histones. Mechanisms that affect the genomic environment include modifications to the DNA itself, absence/presence of histone modifications and histone variants, and also processes involving noncoding

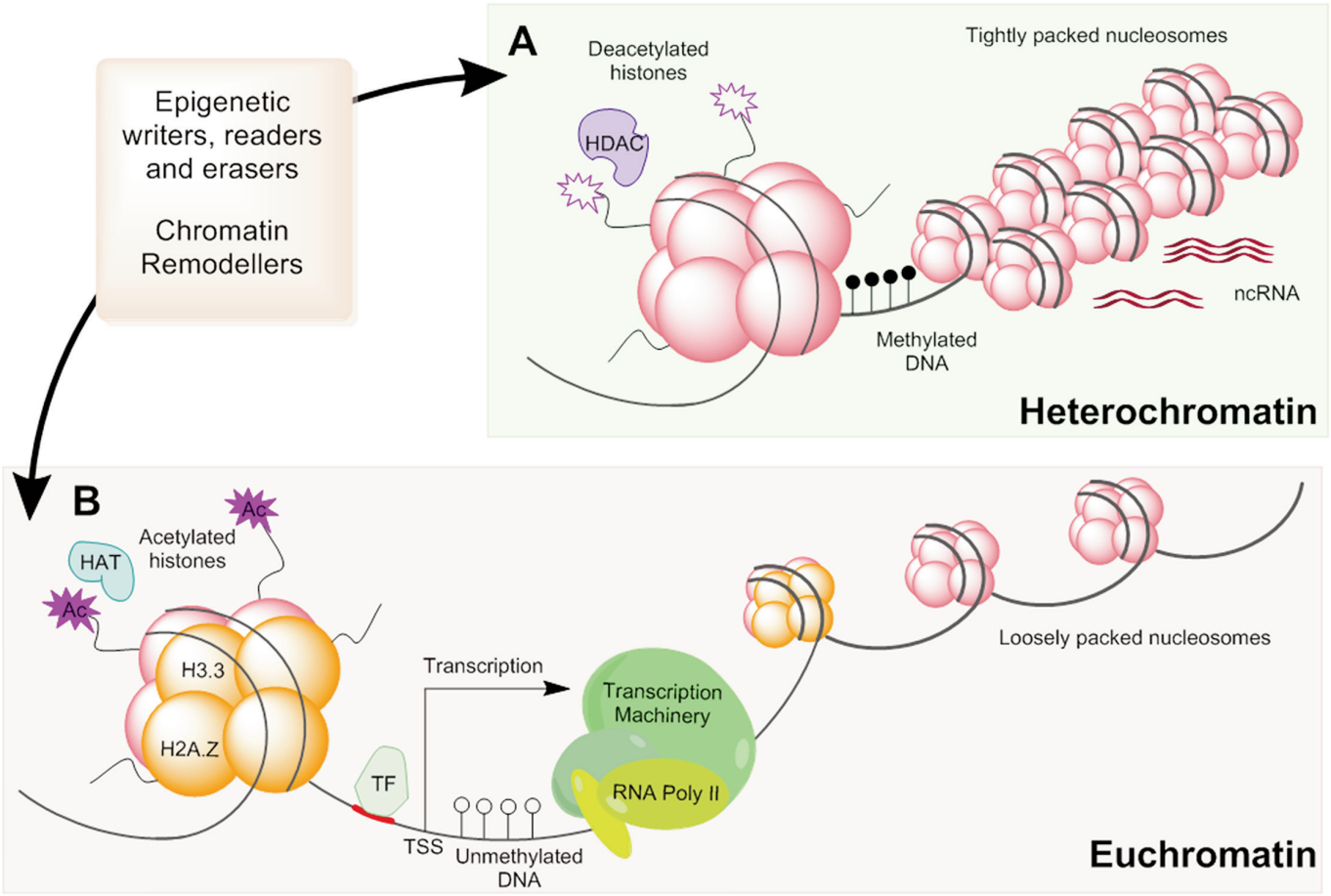


Figure 2. Epigenetic characteristics of heterochromatin and euchromatin. Nucleosome complexes consisting of eight core histone proteins and 147 bp of DNA are configured in higher order structures that regulate the accessibility of DNA to the transcriptional machinery. (A) Heterochromatin is characterized by DNA hypermethylation (filled black circles), de-acetylated histones (clear purple stars) and tightly packed nucleosomes. Noncoding RNA is involved in maintaining heterochromatin structure. (B) Euchromatin contains unmethylated CpGs (unfilled circles), acetylated histones (purple stars) and histone variants H2A.Z and H3.3 and is more accessible to the transcription machinery (green). Writers, readers and erasers of epigenetic modifications work in concert with chromatin remodeling complexes to move and modify nucleosomes, altering chromatin composition. TF, transcription factor; TSS, transcription start site.

RNA and chromatin remodeling complexes (12) (Figure 2). Although there is contention over the extent of heritability of the outcomes of many of these mechanisms and whether they should therefore strictly be regarded as epigenetic mechanisms (9), it is clear that they all influence genome function.

EPIGENETIC MECHANISMS

DNA Methylation

DNA methylation, the addition of a methyl group to a cytosine residue immediately preceding a guanine (CpG dinucleotides), is the most widely studied epigenetic modification. CpG dinucleotides are enriched in clusters called CpG islands, associated with the promoter regions of up to 60% of genes (13). DNA methylation is generally associated with gene silencing, inhibiting gene expression by recruiting proteins that facilitate chromatin condensation and to a lesser extent by physically blocking transcription factor binding (13).

DNA methylation patterns are influenced by the underlying genetic sequence, stochastic changes, environmental factors and other epigenetic mechanisms, resulting in differing methylation patterns across populations, age, tissue and loci (14). DNA methyltransferases (DNMTs) ensure tissue-specific DNA methylation patterns, once established, are maintained through mitosis with high precision and fidelity (13). In contrast, at a global level, methylation is substantially wiped clean during gametogenesis to provide the developing embryo the capacity for totipotency and prevent accumulation of epigenetic changes from previous generations (15). This reprogramming occurs in two waves: first, during pre-implantation

and, second, after primordial germ cell migration, including the removal of imprinted marks (16).

Genomic reprogramming involves both active and passive demethylation, which has long perplexed the scientific community because of their inability to identify the enzyme responsible for this demethylation. Insight into this process has been gained only recently after the discovery of the additional DNA modification, hydroxymethylation, enriched in Purkinje cells in the brain (17) and also embryonic stem cells (18,19). The ten-eleven translocation (TET) family of proteins are responsible for this modification, and there is evidence that it is an intermediate in the process of active demethylation (20). However, the high levels of the modification, particularly in neural cells and during embryonic development, suggests that it may have a yet-to-be-elucidated role in regulating the genome also.

Chromatin Conformation and Structure

Packaging of the vast quantity of DNA into the eukaryotic nucleus is facilitated by the assembly of DNA into nucleosomes followed by their compaction into higher-order structures. This structural organization also plays a fundamental role in regulating accessibility of the DNA to the cellular transcription machinery (12) (Figure 2).

Up to a dozen posttranslational modifications of histone proteins have been reported to date, including methylation, acetylation, phosphorylation and ubiquitination (12), and technological advances have made it possible to map these modifications genome-wide (21). These modifications have together been proposed to form a histone code, which can be interpreted by cellular proteins to specify downstream functions (22). In addition, chromatin structure is altered by the actions of ATP-dependent chromatin remodeling enzymes and the exchange of canonical histones with histone variants. This step creates a highly dynamic, adaptable epigenetic landscape that plays a key role in regulating genome function and provides an interface between the environment and the genome. Although histone modifications can be subject to rapid turnover, there is also evidence that they can be stably inherited during cell division and thus contribute to the maintenance of cellular identity. However, this apparent epigenetic memory may be initiated by other factors such as DNA methylation, noncoding RNA (ncRNA), transcription factors or the underlying DNA sequence (12). Three recent studies point to an important role of genetic variants in determining histone modification patterns (23–25). In these studies, hundreds of variants were associated with changes to histones and gene expression, with the underlying mechanism thought to be altered transcription factor binding.

Noncoding RNA

In recent years, the existence of a complex network of ncRNAs transcribed from the human genome has become apparent. These ncRNAs have regulatory functions and play a key role in the establishment and maintenance of other epigenetic marks (26), with evidence that they constitute a mechanism for transgenerational epigenetic inheritance (27). There is also mounting evidence for the involvement of ncRNAs in disease development, particularly in cancer (28).

GENETIC DETERMINANTS OF EPIGENETIC PATTERNS

Twin Studies

Diminished genetic noise and the innate advantage of perfect age and often sex matching, frequently coupled with similar environmental and socioeconomic upbringing, ensures twin studies play an invaluable role in understanding epigenetic mechanisms (29). The first twin studies examining DNA methylation focused on X-chromosome inactivation, finding that the selection of which X-chromosome is inactivated is not as random as previously thought, but is influenced to a degree by underlying heritable patterns (30). Vickers *et al.* (30) also showed that, with increasing age, there was a greater skew in inactivation patterns, suggesting twins become epigenetically dissimilar with age. This result was later reinforced by several larger studies that found a high degree of epigenetic heritability among MZ twins that decreased with age and that found larger discrepancies between twins that lead different lifestyles (5).

More recently, the advent of methylation array technology has enabled more extensive studies that have found MZ twins to be more epigenetically similar than dizygotic twins. These studies also found that the most heritable CpG sites correlated with functional regions and promoters, indicating these regions are under tighter genetic control (31). Additionally, MZ DNA methylation patterns may be influenced by the chorionicity of the prenatal environment. Perhaps, surprisingly, monochorionicity (a single shared placenta) has been linked to more divergent methylation patterns, yet chorionicity is not always accounted for in MZ twin studies (32). These studies provide evidence for epigenetic differences in genetically identical individuals, suggesting epigenetic metastability independent of genotype can occur in humans, as has been previously shown in mouse models (33). However, they also point to certain regions where genetic influences exert greater control on the epigenome.

Further evidence from studies examining DNA methylation patterns among family members and unrelated individuals found SNPs induce subtle epigenetic variation. One of the first studies to examine the relationship between genetic variants and DNA methylation patterns found evidence for allele-specific methylation (ASM) outside imprinted regions, some linked to allele-specific gene expression (34). Examination of the effect of genetic variants on DNA methylation in a three-generation family and among unrelated individuals found that heterozygous SNPs associated with different methylation patterns (35). This differen-

tial methylation also correlated with gene expression. Globally, genetic variants were found to be more influential than imprinting, and most of the ASM (75%) observed in the family was also present in unrelated individuals, suggesting genotype influences heritable regions of differential methylation.

Other studies estimate around 20% of heterozygous SNPs are linked to ASM (36). Alternative terms have been proposed for these variants, including methylation-associated SNPs (mSNPs) (37), CpG-SNPs (36) and methylation quantitative trait loci (meQTL) (38). Hundreds of these genetic variants have been linked to DNA methylation patterns and, similar to genetic quantitative trait loci, these variants can affect gene expression and phenotype (34,36–38). While SNPs in the vicinity of CpG sites influence methylation levels, the most obvious variant affecting methylation is a mutation at the CpG site itself. Indeed, most of the SNPs linked to ASM are located at the CpG site and are designated meSNPs (36,39). These meSNPs also influence the methylation levels of neighboring CpGs, particularly those close by (within 45 base pairs [bp]) but have also been show to affect CpGs up to 10 kb away (39).

GENETIC VARIANTS CAUSING EPIGENETIC CHANGE IN DISEASE

Although there is no shortage of studies demonstrating a role for epigenetic changes in driving disease, there are now a number of examples in which *cis*-acting variants have been clearly demonstrated to drive the disease-associated epimutations. The term "epimutation" refers to an altered epigenetic state resulting in altered transcriptional activity of a gene. Such *cis*-acting variants have been shown to alter DNA methylation patterns and gene expression in a variety of human tissues (34,36). These changes are likely an indirect result of altered binding of transcription factors, which can either lead to altered recruitment of chromatin modifiers and remodelers and subsequent epigenetic changes or can cause changes in gene expression that predispose the gene to epigenetic silencing (40). Comprehensive genetic analysis facilitated by improved technology has revealed that several diseases involving epigenetic dysfunction have genetic origins.

One such example is the X-linked neurodevelopmental disorder, fragile X syndrome. The genetic defect involves expansion of a trinucleotide repeat sequence (CGG) at the promoter of the fragile X mental retardation gene (*FMR1*), with up to 45 repeats in unaffected individuals and up to 200 in affected individuals (41). The repeat expansion results in methylation of the region and subsequent epigenetic silencing of the gene.

A more distally acting example of sequence variation contributing to disease through altered gene regulation involves the Myc transcription factor. Activation of the Myc transcription factor is suggested to occur in up to 70% of cancers, arising through a range of mechanisms including translocations, gene amplification, enhanced protein translation and stability, or indirectly through signaling pathways that regulate Myc (42). In addition, a number of GWAS have found multiple SNPs on chromosome 8q24 associated with different types of cancer (43). These SNPs occur in a gene desert but have since been found to influence regulation of the Myc oncogene located hundreds of kilobases away. These regions have been shown to contain distal enhancers of the *Myc* gene and highlight that changes to long-range chromatin structures can result in altered gene expression (44).

Imprinting Disorders

Imprinted genes escape the initial phase of epigenetic reprogramming after fertilization, retaining their parental methylation marks and are expressed in a parent of origin manner (45). Around 100 such imprinted genes are currently known, with this number continuing to increase (45). Imprinting provides clear evidence that epigenetic modifications can be inherited through meiosis, and specific diseases ensue when there is an abnormality in either the erasure of existing marks or reestablishment and maintenance of new marks (45). These diseases can result from underlying genetic defects or be due to epimutations (46).

Prader-Willi syndrome and Angelman syndrome are neurological disorders with distinct phenotypes that occur when the same imprinted region on chromosome 15 is nonfunctional. While the vast majority of cases are caused by a single gene mutation or chromosomal deletion, between 1% (Prader-Willi syndrome) and up to 4% in Angelman syndrome are due to an imprinting defect, with the majority of these being primary epimutations, occurring in the absence of DNA sequence mutations (47). Maternally inherited defects lead to Angelman syndrome, whereas paternal imprinting errors lead to Prader-Willi syndrome (47).

Several different disorders result from disruption of epigenetic regulation at the *IGF2/H19* locus, a region in which heritable factors have been shown to have a greater impact on DNA methylation than the accumulation of stochastic and environmental-induced changes (48). DNA methylation at the imprinting control region upstream of the paternal *H19* allele normally silences *H19* expression and activates *IGF2*, whereas the maternal *IGF2* allele is silenced (49). In the Silver-Russell syndrome, a rare developmental disorder, 45% of cases are attributed to an epimutation in the imprinting control region of the paternal *H19* allele (50).

Beckwith-Weidemann syndrome is a congenital overgrowth syndrome with 83% of cases occurring sporadically, of which around 60% are thought to involve epimutations of two imprinting control regions regulating *H19*, *IGF2*, *KCNQ1* and *CDKN1C* (51). Female MZ twins represent a high proportion of Beckwith-Weidemann syndrome cases, and a study of five discordant female MZ twins found that all affected twins had a defect at the imprinted locus *KCNQ1OT1*, which encodes a noncoding RNA that regulates the expression of other imprinted genes (52). Loss of imprinting at the *IGF2/H19* locus is also an

epigenetic cause of around one-third of Wilms tumor, the most common renal cancer in children (49). The defect is also associated with heightened colorectal cancer risk (53) and esophageal squamous cell carcinoma (54).

Lynch Syndrome

Whereas epigenetic changes are well described in disease, particularly cancer, there are now several clear examples of cancer-associated epigenetic changes being genetically driven. These types of interactions can help to explain features of these diseases such as late onset, environmental effects, tissue specificity and also familial associations that do not follow mendelian inheritance patterns. Combining genomic and epigenomic data is also proving to be of value in the search for prognostic signatures in cancer (55), as seen in the Lynch syndrome, an autosomal dominant cancer susceptibility condition. Approximately two-thirds of Lynch syndrome cases result from heterozygous loss-of-function mutations in DNA mismatch repair genes, most commonly mutL homolog 1 (*MLH1*) and mutS protein homolog 2 (*MSH2*) (56).

However, such mutations are not apparent in around one-third of Lynch syndrome cases, some of which (~4% for *MLH1* [56]) can be explained by epimutations in *MLH1* and *MSH2*. These epimutations lead to transcriptional inactivation of the gene, essentially having the same effect as a genomic sequence mutation seen in other Lynch syndrome cases. One possible mechanism underlying these epimutations involves primary DNA methylation changes independent of any sequence change, resulting in labile epimutations, which can be reversed in the germline and are therefore inherited in an unpredictable, nonmendelian manner or not passed on at all.

Alternatively, secondary epimutations may result from underlying sequence changes, including promoter deletions and SNPs (57); for example, the c.-27C>A germline variant in the 5′UTR of the *MLH1* gene has been linked to cancer susceptibility through transcriptional silencing (58). In these cases, the disease follows a more predictable inheritance pattern, since the epimutation is driven by a genetic variant. As yet undiscovered sequence mutations may also be the underlying carcinogenic mechanism in subsets of cancers such as Cowden syndrome, where some individuals have hypermethylation epimutations in the absence of known sequence mutations (59).

Underlying genetic drivers have also been linked to epimutations in sporadic cases of renal cell cancer, where SNPs were associated with promoter hypermethylation of the von Hippel-Lindau (*VHL*) gene in tumor tissue, a gene previously shown to be genetically altered in individuals with the familial form of the cancer (60). Similarly, in colorectal cancer, a C>T point mutation at an enhancer element of the mismatch repair gene O(6)-methylguanine-DNA methyltransferase (*MGMT*) has been linked to aberrant promoter methylation and gene silencing (61). Given the recent technological advances that are enabling integration of genetic, epigenetic and phenotypic data, it is likely that more of examples of diseases resulting from genetic drivers of epigenetic change will be described in the future.

Genetic Mutations in Epigenetic Modifiers

Mutations in genes encoding epigenetic modifiers also contribute to complex diseases, again with cancer being the best described example. Translocations, mutations or overexpression of modifiers such as DNMTs, histone modifying enzymes or chromatin remodeling proteins are well documented in many cancers (62). Aberrant epigenetic modifiers can directly affect regulation of target genes as well as interacting with specific genetic variants of common disease-causing SNPs (63). Whereas the study of other complex diseases are at an earlier stage, there is accumulating evidence to suggest that disruption to epigenetic modifiers plays a role in a range of other diseases, including diabetes, immune diseases and intellectual disabilities such as autism (63).

The simplest example of a genetic mutation driving an epigenetic change and contributing to disease is when the change occurs in a gene encoding an epigenetic modifying enzyme. Some of the more recently described examples include Kabuki syndrome with mutations in the histone methyltransferase gene *MLL2* (64) and Coffin-Siris syndrome involving mutations in SWI/SNF subunit genes (65). Such disorders have been recently reviewed (66). Another straightforward example of this is in the ICF syndrome (immunodeficiency, centromeric instability and facial anomalies), which usually arises because of biallelic mutations in the gene encoding the DNMT3B methylating enzyme (67). The syndrome is a consequence of loss of DNMT activity resulting in genomic hypomethylation. Genomic hypomethylation is a rare disease, which is invariably fatal in early childhood. However, the disease displays phenotypic variability, which is likely due to the differing effects of individual mutations on DNMT3B activity (67).

Rett syndrome, an X-linked neurodevelopmental disorder usually affecting girls, is also due to genetic defects in an epigenetic modifier (in this case, the methyl CpG binding protein MeCP2) (68). The disease displays delayed onset, with children developing normally until 1–2 years of age, when they present with progressive neurological dysfunction. The largely neurological phenotype of this disease is likely a result of the requirement for tight regulation of a number of important neural targets of MeCP2 (69). Variability in disease phenotype is likely a function of the range of mutations that give rise to the disease as well as an effect of X-inactivation skewing.

THE EPIGENOME: AN INTERFACE BETWEEN THE GENOME AND THE ENVIRONMENT

While the underlying genome plays a role in determining epigenetic profiles, stochastic factors and environmental cues including diet, exercise and toxins bring

about subsequent changes in the epigenome, as previously reviewed comprehensively (70). There is evidence that at least some of these changes can then be passed down through meiosis as transgenerational epigenetic inheritance (26,71,72). Whereas the evidence for this and mechanisms involved are beyond the scope of this article, they have recently been comprehensively reviewed (11).

Stochastic Factors

It is now thought that randomly induced epigenetic patterns may also contribute to variation in development and aging as well as providing a possible mechanism for the rapid selection of epigenotypes in response to environmental pressures. X-chromosomal inactivation is a classic example of how epigenetic profiles can be regulated by stochastic factors. These factors may also explain discordance between MZ twins (73).

While the effect of environment on epigenetic profiles, particularly DNA methylation, is widely acknowledged, stochastic changes may in fact be more common than environmentally induced changes, with a study examining 4,000 human genes, finding 300 to have random monoallelic expression (74). Epigenetic stochasticity can be defined as a combination of epigenetic variation in the germline and somatic instability. Similar to Richards' "facilitated epigenetic variation" model (75), Feinberg and Irizarry's "inherited stochastic variation model" proposes genetic sequence variation underlies the propensity for epigenetic variation, since certain DNA sequences are not only directly responsible for particular traits but also increase natural methylation variation for that trait (76). Various stochastic and environmental factors then influence DNA methylation at these variably methylated regions, resulting in increased phenotypic differences, which are then acted on by Darwinian selection in a similar manner to selection pressures affecting purely genetic traits. Subsequent studies found the sites of greatest DNA methylation variability in colon cancer corresponded to the sites of greatest variability in other cancers, including lung, breast and ovarian cancers, with these sites normally having distinct tissue-specific DNA methylation patterns (77). Thus, heritable DNA methylation variation could provide some contribution to the unexplained heritable genetic component of common complex diseases.

THE PROMISE OF EPIGENETIC THERAPY

Whereas genetic mutations and chromosomal defects permanently alter the genome, epigenetic alterations, whether driven by changes to the underlying genome or by environmental or stochastic influences, can potentially be pharmacologically reversed or modified, providing the promise of restoring gene function, altered as a result of epigenetic changes in disease. There is currently considerable interest in the development and clinical translation of pharmacological agents that target either the writers or the readers of the epigenetic code. Because these are mainly enzymes, they provide an easier target than other gene regulators such as transcription factors. For obvious reasons, the use of such agents is at the most progressed stage in the treatment of cancers, with two DNMT and two HDAC inhibiting compounds approved for cancer treatment in the United States and numerous others in clinical trials (78) (Figure 3).

In 2004, 5-azacytidine (aza-C) became the first U.S. Food and Drug Administration (FDA)-approved epigenetic drug when it was approved for treatment of myelodysplastic syndrome (79). An analog of cytidine, the nucleoside is incorporated into nucleic acid of dividing cells with a preference for RNA over DNA. The presence of nitrogen at carbon-5 blocks the addition of a methyl group by DNMTs, preventing the methylation of the DNA after cell division. The bound DNMTs, unable to detach due to the nitrogen, form permanent adducts with the nucleic acid resulting in functional depletion of DNMT from the cell (Figure 3C). In addition, DNA replication is blocked (80), and the DNA is functionally compromised, activating the p53 damage pathway, leading to degradation of the DNA (81) (Figure 3D).

Aza-2-deoxycytidine (decitabine) is the deoxy form of aza-C and is solely incorporated into DNA, avoiding the indirect effects on RNA and protein synthesis of 5-aza-C. Approved in 2006 for treatment of myelodysplastic syndrome, it is the only other demethylating agent currently approved by the FDA. Other cytidine analogs such as zebularine and 5-fluoro-deoxycytidine are in clinical trials, as are direct DNMT inhibitors including procaine, procanamide and hydralazine (78). The two other FDA-approved epigenetic drugs fall under the umbrella of histone deacetylase inhibitors. Both vorinostat in 2006 and romidepsin in 2009 were approved for treatment of cutaneous T-cell lymphoma (78) (Figure 3A). Interest in targeting epigenetic modifiers in cancer and other diseases continues to grow, with a wide range of targets now being explored in both preclinical and clinical trials. For example, inhibitors of several histone methyltransferases, including EZH2 and DOT1L are also in preclinical trials for certain lymphomas and leukemias, respectively (78) (Figure 3B).

Combining various epigenetic therapies may prove to be the most effective strategy because of the high degree of biological interaction between DNA methylation, histone modifications and chromatin remodeling complexes. Indeed, clinical trials examining the synergistic action of these therapies are promising (82). These therapies are also most likely to be effective when combined with conventional cancer treatments (78). Epigenetic pharmaceuticals are still in their infancy, and clearly more is to be learned about the underlying mechanisms determining epigenetic states, since only some cancers can be reprogrammed to a normal state and demethylating agents and HDAC inhibitors are unable to bring about permanent expression changes. This result is particularly true if the abnormal epigenetic state is driven by underlying ge-

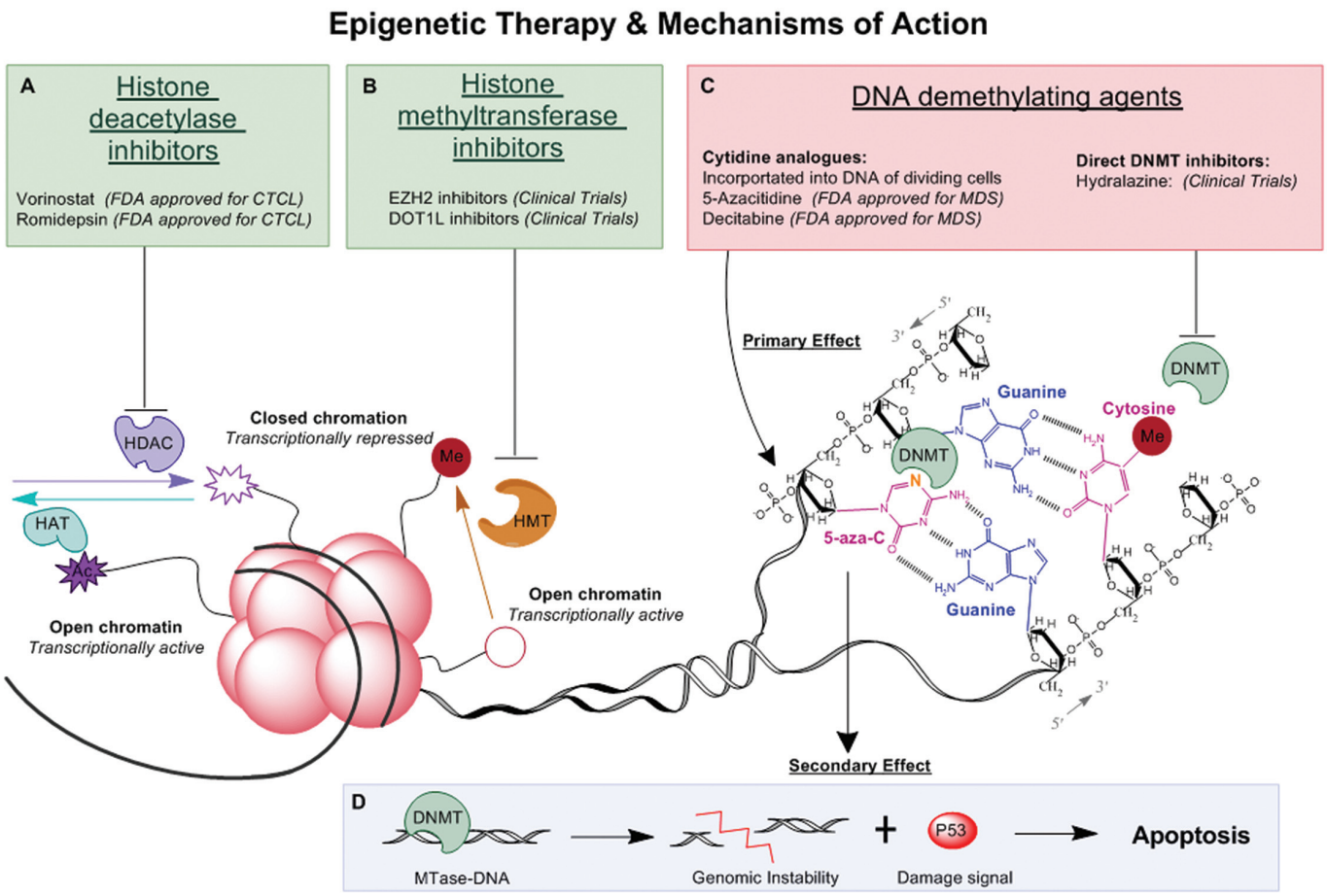


Figure 3. Advances in epigenetic therapy. Epigenetic modifiers and modifications provide targets for therapeutic intervention in disease. (A) Two histone deacetylase inhibitors are FDA approved for treatment of subtypes of leukemia. (B) Inhibitors of a range of epigenetic modifiers, including histone methyltransferase enzymes, are in preclinical trials. (C) DNA demethylating agents decrease genomic methylation, restoring aberrantly silenced gene expression by acting directly to inhibit methylating enzymes or as cytidine analogs are incorporated into nucleic acid of dividing cells, preventing methylation. (D) Cytidine analogs have the secondary effect of activating the p53 damage pathway and inducing apoptosis.

netic factors and would therefore require ongoing pharmacological intervention to prevent reversion of the epigenetic state.

Disease screening and diagnosis may also be vastly improved with the inclusion of epigenetic information. In 2012, Teschendorff *et al.* (83,84) were able to predict risk of cervical neoplasia 3 years before morphological changes by examining DNA methylation variability. Feinberg and Irizarry (85) suggest that if such tests are used to identify subgroups for further traditional follow-up screening (for example, mammogram, colonoscopy), the positive predictive value of these more invasive and expensive tests could rise to over 90% and greatly reduce cancer deaths.

CONCLUDING REMARKS AND FUTURE PERSPECTIVES

Many questions remain regarding the underlying mechanisms that determine epigenetic patterns. Rapid improvements in technology and our increasing ability to effectively analyze and interpret the immense quantities of data produced are allowing the generation of comprehensive epigenomic maps and characterization of the differences in epigenetic state between individuals and the changes that occur during development, aging and disease processes. This information is also aiding in our understanding of the underlying mechanisms involved, including the relative contribution of environmental influences, stochastic factors and genetic variants. Numerous studies across a range of tissue types and populations are providing strong evidence for a key role for genetic variants in establishing inherited methylation patterns.

Over recent years, we have gained considerable insight into the role of genetic variants and epigenetic change in diseases. Attention is now turning to understanding the interactions between genetic and epigenetic factors and their concerted roles in disease processes. This

approach is providing advances in disease prevention, diagnosis and surveillance. It is also offering hope that a heightened understanding of how inherited factors regulate gene expression through epigenetic mechanisms will provide more personalized diagnostic tools and effective treatments for complex disease.

ACKNOWLEDGMENTS

The authors were supported by grants from the David Collins Leukaemia Foundation and The Cancer Council Tasmania. JL Dickinson is an Australian Research Council Future Fellow. E Cazaly was supported by a scholarship from the Royal Hobart Hospital Cancer Auxiliary.

The authors acknowledge the many researchers whose work has contributed to this field, but they are not specifically cited here because of space and referencing constraints.

DISCLOSURE

The authors declare that they have no competing interests as defined by *Molecular Medicine*, or other interests that might be perceived to influence the results and discussion reported in this paper.

REFERENCES

1. Botstein D, Risch N. (2003) Discovering genotypes underlying human phenotypes: past successes for mendelian disease, future approaches for complex disease. *Nat. Genet.* 33:228–37.
2. Eichler EE, *et al.* (2010) Missing heritability and strategies for finding the underlying causes of complex disease. *Nat. Rev. Genet.* 11:446–50.
3. Hindorff LA, *et al.* (2009) Potential etiologic and functional implications of genome-wide association loci for human diseases and traits. *Proc. Natl. Acad. Sci. U. S. A.* 106:9362–7.
4. Furey TS, Sethupathy P. (2013) Genetics: genetics driving epigenetics. *Science.* 342:705–6
5. Steves CJ, *et al.* (2012) Ageing, genes, environment and epigenetics: what twin studies tell us now, and in the future. *Age Ageing.* 41:581–6.
6. Laird PW (2010) Principles and challenges of genome-wide DNA methylation analysis. *Nat. Rev. Genet.* 11:191–203.
7. Montavon C, *et al.* (2012) Prognostic and diagnostic significance of DNA methylation patterns in high grade serous ovarian cancer. *Gynecol. Oncol.* 124:582–8.
8. Waddington CH. (2012) The epigenotype. *Int. J. Epidemiol.* 41:10–3.
9. Berger SL, *et al.* (2009) An operational definition of epigenetics. *Genes Dev.* 23:781–3.
10. Skinner MK. (2011) Environmental epigenetic transgenerational inheritance and somatic epigenetic mitotic stability. *Epigenetics.* 6:838–42.
11. Pembrey M, *et al.* (2014) Human transgenerational responses to early-life experience: potential impact on development, health and biomedical research. *J. Med. Genet.* 51:563–72.
12. Dawson MA, Kouzarides T. (2012) Cancer epigenetics: from mechanism to therapy. *Cell.* 150:12–27.
13. Cedar H, Bergman Y. (2012) Programming of DNA methylation patterns. *Annu. Rev. Biochem.* 81:97–117.
14. ENCODE Project Consortium. (2012) An integrated encyclopedia of DNA elements in the human genome. *Nature.* 489:57–74.
15. Seisenberger S, *et al.* (2012) The dynamics of genome-wide DNA methylation reprogramming in mouse primordial germ cells. *Mol. Cell.* 48:849–62.
16. Sasaki H, Matsui Y. (2008) Epigenetic events in mammalian germ-cell development: reprogramming and beyond. *Nat. Rev. Genet.* 9:129–40.
17. Kriaucionis S, Heintz N. (2009) The nuclear DNA base 5-hydroxymethylcytosine is present in Purkinje neurons and the brain. *Science.* 324:929–30.
18. Ito S, *et al.* (2010) Role of Tet proteins in 5mC to 5hmC conversion, ES-cell self-renewal and inner cell mass specification. *Nature.* 466:1129–33.
19. Cimmino L, *et al.* (2011) TET family proteins and their role in stem cell differentiation and transformation. *Stem Cell.* 9:193–204.
20. Hackett JA, *et al.* (2013) Germline DNA demethylation dynamics and imprint erasure through 5-hydroxymethylcytosine. *Science.* 339:448–52.
21. Barski A, *et al.* (2007) High-resolution profiling of histone methylations in the human genome. *Cell.* 129:823–37.
22. Wang Y, *et al.* (2004) Beyond the double helix: reading and writing the histone code. *Novartis Found. Symp.* 259:3–17.
23. McVicker G, *et al.* (2013) Identification of genetic variants that affect histone modifications in human cells. *Science.* 342:747–9.
24. Kilpinen H, *et al.* (2013) Coordinated effects of sequence variation on DNA binding, chromatin structure, and transcription. *Science.* 342:744–7.
25. Kasowski M, *et al.* (2013) Extensive variation in chromatin states across humans. *Science.* 342:750–2.
26. Daxinger L, Whitelaw E. (2012) Understanding transgenerational epigenetic inheritance via the gametes in mammals. *Nat. Rev. Genet.* 13:153–62.
27. Liebers R, *et al.* (2014) Epigenetic regulation by heritable RNA. *PLoS. Genet.* 10:e1004296.
28. Kasinski AL, Slack FJ. (2011) Epigenetics and genetics. MicroRNAs en route to the clinic: progress in validating and targeting microRNAs for cancer therapy. *Nat. Rev. Cancer.* 11:849–64.
29. Bell JT, Spector TD. (2011) A twin approach to unraveling epigenetics. *Trends Genet.* 27:116–25.
30. Vickers MA, *et al.* (2001) Assessment of mechanism of acquired skewed X inactivation by analysis of twins. *Blood.* 97:1274–81.
31. Kaminsky ZA, *et al.* (2009) DNA methylation profiles in monozygotic and dizygotic twins. *Nat. Genet.* 41:240–5.
32. Race JP, *et al.* (2006) Chorion type, birthweight discordance and tooth-size variability in Australian monozygotic twins. *Twin Res. Hum. Genet.* 9:285–91.
33. Morgan HD, *et al.* (1999) Epigenetic inheritance at the agouti locus in the mouse. *Nat. Genet.* 23:314–8.
34. Kerkel K, *et al.* (2008) Genomic surveys by methylation-sensitive SNP analysis identify sequence-dependent allele-specific DNA methylation. *Nat. Genet.* 40:904–8.
35. Gertz J, *et al.* (2011) Analysis of DNA methylation in a three-generation family reveals widespread genetic influence on epigenetic regulation. *PLoS. Genet.* 7:e1002228.
36. Shoemaker R, *et al.* (2010) Allele-specific methylation is prevalent and is contributed by CpG-SNPs in the human genome. *Genome Res.* 20:883–9.
37. Zhang D, *et al.* (2010) Genetic control of individual differences in gene-specific methylation in human brain. *Am. J. Human Genet.* 86:411–9.
38. Bell JT, *et al.* (2011) DNA methylation patterns associate with genetic and gene expression variation in HapMap cell lines. *Genome Biol.* 12:R10.
39. Zhi D, *et al.* (2013) SNPs located at CpG sites modulate genome-epigenome interaction. *Epigenetics.* 8:802–6.
40. Hesson LB, *et al.* (2010) Epimutations and cancer predisposition: importance and mechanisms. *Curr. Opin. Genet. Dev.* 20:290–8.
41. Fu Y-H, *et al.* (1991) Variation of the CGG repeat at the fragile X site results in genetic instability: resolution of the Sherman paradox. *Cell.* 67:1047–58.
42. Meyer N, Penn LZ. (2008) Reflecting on 25 years with MYC. *Nat. Rev. Cancer.* 8:976–90.
43. Haiman CA, *et al.* (2007) Multiple regions within 8q24 independently affect risk for prostate cancer. *Nat. Rev. Cancer.* 39:638–44.
44. Sotelo J, *et al.* (2010) Long-range enhancers on 8q24 regulate c-Myc. *Proc. Natl. Acad. Sci. U. S. A.* 107:3001–5.
45. Ferguson-Smith AC. (2011) Genomic imprinting: the emergence of an epigenetic paradigm. *Nat. Rev. Genet.* 12:565.
46. Banno K, *et al.* (2012) Epimutation and cancer: a new carcinogenic mechanism of Lynch syndrome (Review). *Int. J. Oncol.* 41:793–7.
47. Buiting K, *et al.* (2003) Epimutations in Prader-Willi and Angelman syndromes: a molecular study of 136 patients with an imprinting defect. *Am. J. Hum. Genet.* 72:571–7.
48. Heijmans BT, *et al.* (2007) Heritable rather than age-related environmental and stochastic factors dominate variation in DNA methylation of the human IGF2/H19 locus. *Hum. Mol. Genet.* 16:547–54.
49. Ludgate JL, *et al.* (2013) Global demethylation in loss of imprinting subtype of Wilms tumor. *Genes Chromosomes Cancer.* 52:174–84.
50. Fuke T, *et al.* (2013) Molecular and clinical studies in 138 Japanese patients with Silver-Russell syndrome. *PLoS One.* 8:e60105.

51. Murrell A, *et al.* (2004) An association between variants in the IGF2 gene and Beckwith-Wiedemann syndrome: interaction between genotype and epigenotype. *Hum. Mol. Genet.* 13:247–55.
52. Weksberg R, *et al.* (2002) Discordant KCNQ1OT1 imprinting in sets of monozygotic twins discordant for Beckwith-Wiedemann syndrome. *Hum. Mol. Genet.* 11:1317–25.
53. Cui H, *et al.* (2003) Loss of IGF2 imprinting: a potential marker of colorectal cancer risk. *Science.* 299:1753–5.
54. Murata A, *et al.* (2014) IGF2 DMR0 methylation, loss of imprinting, and patient prognosis in esophageal squamous cell carcinoma. *Ann. Surg. Oncol.* 21:1166–74.
55. Yi JM, *et al.* (2011) Genomic and epigenomic integration identifies a prognostic signature in colon cancer. *Clin. Cancer Res.* 17:1535–45.
56. Ward RL, *et al.* (2013) Identification of constitutional MLH1 epimutations and promoter variants in colorectal cancer patients from the Colon Cancer Family Registry. *Genet. Med.* 15:25–35.
57. Hitchins MP, Lynch HT. (2014) Dawning of the epigenetic era in hereditary cancer. *Clin. Genet.* 85:413–6.
58. Hitchins MP, *et al.* (2011) Dominantly inherited constitutional epigenetic silencing of MLH1 in a cancer-affected family is linked to a single nucleotide variant within the 5′UTR. *Cancer Cell.* 20:200–13.
59. Bennett KL, *et al.* (2010) Germline epigenetic regulation of KILLIN in Cowden and Cowden-like syndrome. *JAMA.* 304:2724–31.
60. Moore LE, *et al.* (2011) Von Hippel-Lindau (VHL) inactivation in sporadic clear cell renal cancer: associations with germline VHL polymorphisms and etiologic risk factors. *PLoS Genet.* 7:e1002312.
61. Ogino S, *et al.* (2007) MGMT germline polymorphism is associated with somatic MGMT promoter methylation and gene silencing in colorectal cancer. *Carcinogenesis.* 28:1985–90.
62. You JS, Jones PA. (2012) Cancer genetics and epigenetics: two sides of the same coin? *Cancer Cell.* 22:9–20.
63. Bjornsson H. (2004) An integrated epigenetic and genetic approach to common human disease. *Trends Genet.* 20:350–8.
64. Ng SB, *et al.* (2010) Exome sequencing identifies MLL2 mutations as a cause of Kabuki syndrome. *Nat. Genet.* 42:790–3.
65. Tsurusaki Y, *et al.* (2012) Mutations affecting components of the SWI/SNF complex cause Coffin-Siris syndrome. *Nat Genet.* 44:376–8.
66. Berdasco M, Esteller M. (2013) Genetic syndromes caused by mutations in epigenetic genes. *Hum. Genet.* 132:359–83.
67. Hansen RS, *et al.* (1999) The DNMT3B DNA methyltransferase gene is mutated in the ICF immunodeficiency syndrome. *Proc. Natl. Acad. Sci. U.S.A.* 96:14412–7.
68. Liyanage VRB, Rastegar M. (2014) Rett syndrome and MeCP2. *Neuromol. Med.* 16:231–64.
69. Chen WG, *et al.* (2003) Derepression of BDNF transcription involves calcium-dependent phosphorylation of MeCP2. *Science.* 302:885–9.
70. Feil R, Fraga MF. (2012) Epigenetics and the environment: emerging patterns and implications. *Nat. Rev. Genet.* DOI: 10.1038/nrg3142.
71. Veenendaal MVE, *et al.* (2013) Transgenerational effects of prenatal exposure to the 1944–45 Dutch famine. *BJOG.* 120:548–53.
72. Guerrero-Bosagna C, Skinner MK. (2011) Environmentally induced epigenetic transgenerational inheritance of phenotype and disease. *Mol. Cell. Endocrinol.* 354:1–6.
73. Petronis A. (2010) Epigenetics as a unifying principle in the aetiology of complex traits and diseases. *Nature.* 465:721–7.
74. Gimelbrant A, *et al.* (2007) Widespread monoallelic expression on human autosomes. *Science.* 318:1136–40.
75. Richards EJ. (2006) Inherited epigenetic variation: revisiting soft inheritance. *Nat. Rev. Genet.* 7:395–401.
76. Feinberg AP, Irizarry RA. (2010) Evolution in health and medicine Sackler colloquium: stochastic epigenetic variation as a driving force of development, evolutionary adaptation, and disease. *Proc. Natl. Acad. Sci. U. S. A.* 107:1757–64.
77. Hansen KD, *et al.* (2011) Increased methylation variation in epigenetic domains across cancer types. *Nat. Genet.* 43:768–75.
78. Bojang P Jr, Ramos KS. (2014) The promise and failures of epigenetic therapies for cancer treatment. *Cancer Treat. Rev.* 40:153–69.
79. Kaminskas E, *et al.* (2005) FDA drug approval summary: azacitidine (5-azacytidine, Vidaza™) for injectable suspension. *Oncologist.* 10:176–82.
80. Kuo HK, *et al.* (2007) 5-Azacytidine-induced methyltransferase-DNA adducts block DNA replication in vivo. *Cancer Res.* 67:8248–54.
81. Karpf AR, *et al.* (2001) Activation of the p53 DNA damage response pathway after inhibition of DNA methyltransferase by 5-aza-2′-deoxycytidine. *Mol. Pharmacol.* 59:751–7.
82. Garcia-Manero G, *et al.* (2006) Phase 1/2 study of the combination of 5-aza-2′-deoxycytidine with valproic acid in patients with leukemia. *Blood.* 108:3271–9.
83. Teschendorff AE, Widschwendter M. (2012) Differential variability improves the identification of cancer risk markers in DNA methylation studies profiling precursor cancer lesions. *Bioinformatics.* 28:1487–94.
84. Teschendorff AE, *et al.* (2012) Epigenetic variability in cells of normal cytology is associated with the risk of future morphological transformation. *Genome Med.* 4:24.
85. Feinberg AP. (2014) Epigenetic stochasticity, nuclear structure and cancer: the implications for medicine. *J. Intern. Med.* 276:5–11.

Cite this article as: Cazaly E, Charlesworth J, Dickinson JL, Holloway AF. (2015) Genetic determinants of epigenetic patterns: providing insight into disease. *Mol. Med.* 21:400–9.

```
## Chapter.2 Appendix -- Analysis pipeline for initial quality
control of Methylation Array Data ##


#### Overview ####
# 1. Read in data
# 2. Remove failed samples
# 3. Background correction & Control normalisation
# 4. Convert from RGset to MethylSet
# 5. Remove sex chr and normalise separately then recombine.  This
is tricky for Methylumi objects as annotation seems impossible
#  ** Below 3 steps performed in next chapter --> normalisation **
### 5. Normalisation: swan, QN, BMIQ, Dasen
### 6. Remove failed probes based on detP- nonsignificant signal
compared to background 0.05 threshold
### 7. Batch effect correction
#########################################################################
##############################


################
# 1. Load data #
################
library(minfi)
baseDir <- file.path("/Users/ecazaly/Desktop/PhD_Analysis/450K/Raw
data/Combined data IDAT files")
list.files(baseDir)
targets <- read.450k.sheet(baseDir)
# alternatively
# load("/Users/ecazaly/Dropbox/Analysis/workingDROP/RGsetE.RData")
RGsetE=read.450k.exp(base=baseDir,targets=targets, extended=TRUE)
RGsetE<- updateObject(RGsetE)

############################
# 2. Remove failed samples #
############################
bad=c(9,17,22,24,27,29,40,42)
RGsetE2=RGsetE[,-c(bad)]
# which are the poor quality samples?
RGsetE[,bad]@phenoData$Sample_Name
# [1] "pc11-233a"     "PC11-213a"     "pc72-291a"
# [4] "pc72-291b"     "PC9-4T (AH3)"  "PC9-12T (AH5)"
# [7] "PC11-233b"     "PC11-213b"

densityPlot(RGsetE2,sampGroups=RGsetE2@phenoData$Batch, pal =
rainbow(3), xlim=c(-.1,1.2),ylim=c(0,3.6), xlab="Beta",
main="RGsetE2")
mdsPlot(RGsetE2, numPositions=1000, sampNames=RGsetE2@phenoData
$Sample_Name, sampGroups=RGsetE2@phenoData$Batch,
        xlim=c(-4.5,9), ylim=c(-5,8),legendNCol=5, legendPos =
"bottom", main="MDS RGsetE2")

#################################################
# 3. Background correction, control normalisation #
```

```
##################################################
#preproIllumina will return a methylSet or bgcorrect.illumina and
normalize.illumina.control will return an RGset

RGsetE2_bgcorrect=bgcorrect.illumina(RGsetE2)
densityPlot(RGsetE2_bgcorrect,sampGroups=RGsetE2_bgcorrect@phenoData
$Batch, pal = rainbow(3), xlim=c(-.1,1.2),ylim=c(0,3.6),
xlab="Beta", main="RGsetE2_bgcorrect")
mdsPlot(RGsetE2_bgcorrect, numPositions=1000,
sampNames=RGsetE2_bgcorrect@phenoData$Sample_Name,
sampGroups=RGsetE2_bgcorrect@phenoData$Batch,
        xlim=c(-8.5,5), ylim=c(-5,8),legendNCol=5, legendPos =
"bottom", main="MDS RGsetE2_bgcorrect")

RGsetE2_control_norm=normalize.illumina.control(RGsetE2_bgcorrect,
reference = 1)  # try with diff refs
RGsetE2_control_norm2=normalize.illumina.control(RGsetE2_bgcorrect,
reference = 10)
RGsetE2_control_norm3=normalize.illumina.control(RGsetE2_bgcorrect,
reference = 20)
RGsetE2_control_norm4=normalize.illumina.control(RGsetE2_bgcorrect,
reference = 40)
densityPlot(RGsetE2_control_norm,sampGroups=RGsetE2_control_norm@phe
noData$Batch, pal = rainbow(3), xlim=c(-.1,1.2),ylim=c(0,3.6),
xlab="Beta", main="control norm ref1")
densityPlot(RGsetE2_control_norm2,sampGroups=RGsetE2_control_norm2@p
henoData$Batch, pal = rainbow(3), xlim=c(-.1,1.2),ylim=c(0,3.6),
xlab="Beta", main="control norm ref2")
densityPlot(RGsetE2_control_norm3,sampGroups=RGsetE2_control_norm3@p
henoData$Batch, pal = rainbow(3), xlim=c(-.1,1.2),ylim=c(0,3.6),
xlab="Beta", main="control norm ref3")
densityPlot(RGsetE2_control_norm4,sampGroups=RGsetE2_control_norm4@p
henoData$Batch, pal = rainbow(3), xlim=c(-.1,1.2),ylim=c(0,3.6),
xlab="Beta", main="control norm ref4")
# These plots look identical
mdsPlot(RGsetE2_control_norm, numPositions=1000,
sampNames=RGsetE2_control_norm@phenoData$Sample_Name,
sampGroups=RGsetE2_control_norm@phenoData$Batch,
        xlim=c(-10.5,4.5), ylim=c(-8.5,5),legendNCol=5, legendPos =
"bottom", main="MDS RGsetE2_control_norm")
mdsPlot(RGsetE2_control_norm2, numPositions=1000,
sampNames=RGsetE2_control_norm2@phenoData$Sample_Name,
sampGroups=RGsetE2_control_norm2@phenoData$Batch,
        xlim=c(-10.5,4.5), ylim=c(-8.5,5),legendNCol=5, legendPos =
"bottom", main="MDS RGsetE2_control_norm2")
mdsPlot(RGsetE2_control_norm3, numPositions=1000,
sampNames=RGsetE2_control_norm3@phenoData$Sample_Name,
sampGroups=RGsetE2_control_norm3@phenoData$Batch,
        xlim=c(-5,10), ylim=c(-8.5,4),legendNCol=5, legendPos =
"bottom", main="MDS RGsetE2_control_norm3")
mdsPlot(RGsetE2_control_norm4, numPositions=1000,
sampNames=RGsetE2_control_norm4@phenoData$Sample_Name,
sampGroups=RGsetE2_control_norm4@phenoData$Batch,
        xlim=c(-10.5,4.5), ylim=c(-8.5,5),legendNCol=5, legendPos =
```

```
"bottom", main="MDS RGsetE2_control_norm4")
# As do these, although the 3rd one inverts


#########################################################################
#########
# 4. Convert from RGset to MethylSet without normalisation or p-val
removal #
#########################################################################
#########
Raw_preprocess <- preprocessRaw(RGsetE2_control_norm)
# MethylSet (storageMode: lockedEnvironment)
# 485512 features, 52 samples  - this has lost 136887 sites


#########################################################################
########
# 5. Remove X&Y chr and normalise these separately if using female
samples #
#########################################################################
########
Annotated=getAnnotation(Raw_preprocess, what="everything",
drop=FALSE)  # this is a dataframe, could be an issue downstream
dim(Annotated) # 485512     33  - cuts it down to the same number of
probes in Raw_preprocess.
chrs=Annotated@listData$chr
Autosomes=Raw_preprocess[chrs!="chrX" & chrs !="chrY"]  # Methylset
dim(Autosomes) #  473864       52
dim(Raw_preprocess) # 485512       52  : have removed 11648 sex chr
Annotated_Autosomes=getAnnotation(Autosomes, what="everything",
drop=FALSE)  #dataframe
 #check:
dim(Annotated_Autosomes)  # [1] 473864     33
which(Annotated_Autosomes[,1]=="chrY") #integer(0)
which(Annotated_Autosomes[,1]=="chrX") #integer(0)

# normalise these automsomes separately to sex chr then somehow
recombine

Raw_Male <- Raw_preprocess[,Raw_preprocess@phenoData$Sex=="M"]
Male_chr <- Raw_Male[chrs=="chrY" | chrs=="chrX"]
dim(Male_chr) #  Features  Samples   11648       43
Raw_Female <- Raw_preprocess[,Raw_preprocess@phenoData$Sex=="F"]
Female_chr <- Raw_Female[chrs=="chrX"]
dim(Female_chr) # Features  Samples 11232       9
```

Chapter.2 Appendix.2 Analysis Pipeline for Omni2.5 genotyping array quality control using PLINK

```
# navigate to correct directory on HPC :  cd ~/Data/Omni25/pipeline_omni

# ------ Add phenotype data -------
# To add phenotypes; transpose to t.ped and t.fam files and then manually edit t.fam files in VI or nano
plink --noweb --file All_Chips_1_5Cutoff_noReps --transpose --recode --out cutoff_15
# in nano modify final column with 2 (A), 1 (UA), 0 (NA) and then exit, will ask to save on exiting
nano cutoff_15.tfam  #
# at this point there are 2,391739 SNPs
# 27534 heterozygous haploid genotypes; set to missing
# haploid chromosomes are only counted once (i.e. male X and Y chromosome SNPs and all MT SNPs).
# genotypes have been set to missing if they are not valid (female Y genotype, heterozygous haploid chromosome)
# 192 SNPs with no founder genotypes observed


# ------- QC per individual -------

# 2,391739 SNPs
# 1. Check gender assignment
plink --noweb --tfile cutoff_15 --check-sex --out sexcheck_15
#look at the file, 5th column gives status as OK or otherwise.  All ok for my samples
less sexcheck_15.sexcheck


# 2. Missingness: remove indivs with >10% missing genotypes
# 2,391739 SNPs
plink --noweb --tfile cutoff_15 --mind 0.1 --make-bed --out cutoff_15_clean
# Check plink log. file to see if any excluded.  No all left in, so must have less than 10% which agrees
# with genomeStudio as call rates only down to 96% and mostly 99%


# 3. Mendelian errors - can't use this as have no trios
#plink --noweb --bfile cutoff_15_clean --mendel --out cutoff_15_clean_mendel
###Currently, PLINK only scans full trios for Mendel errors. Families with fewer than 2 parents in the dataset will not be tested.###
```

```
# ------- QC per SNP -------

# 1. Remove SNPs with high rate of missing genotype calls -more than
10% missing genotypes... this is actually set at 5%
# 2,391739 SNPs
plink --noweb --bfile cutoff_15_clean --geno 0.05 --make-bed --out
cutoff_15_clean2
# 61217 SNPs failed missingness test ( GENO > 0.05 ), so now have
2330522 SNPs
# 2330522 SNPs

# 2. Remove SNPs out of Hardy-Weinberg equilibrium
# 2330522 SNPs
###plink --noweb --bfile cutoff_15_clean2 --hardy --out
cutoff_15_clean2_hardy
# 335 markers to be excluded based on HWE test ( p <= 0.001 )
        #1303 markers failed HWE test in cases
        #335 markers failed HWE test in controls
        #Total genotyping rate in remaining individuals is 0.993477
# 2330187 SNPs
# is this too stringent? should I perhaps use 0.0001 which doesn't
exclude any?

plink --noweb --bfile cutoff_15_clean2 --hwe 1E-4 --hardy --make-bed
--out cutoff_15_clean_hwe_1E-4
# 0 markers to be excluded based on HWE test ( p <= 0.0001 )
        # 419 markers failed HWE test in cases
        # 0 markers failed HWE test in controls
        # Total genotyping rate in remaining individuals is
0.993679
# 2330522 SNPs

###plink --noweb --bfile cutoff_15_clean2 --hardy2 --out
cutoff_15_clean2_hardy2
# --hardy2 is a lot more stringent and less accurate for rare
genotypes, it takes out about 10 thousand more
        #10522 markers to be excluded based on HWE test ( p <=
0.001 )
        #15887 markers failed HWE test in cases
        #10522 markers failed HWE test in controls
        #Total genotyping rate in remaining individuals is 0.993477

# check in R
hwe2 <- read.delim("cutoff_15_clean_hwe_1E-4.hwe", header=T,
as.is=T, sep="")
str(hwe2)
hweALL2 <- hwe2[which(hwe2$TEST=="ALL"),]  # get rid of the rows
that specify AFF, UNAFF
# order by p-val, p-vals are generated per SNP depending on the
deviation from HWE, low p-values indicate a
# SNP is out of HWE.
hweOrderALL2 <- hweALL2[order(hweALL2$P),]
hwe2_less10_5 <- hweALL2[which(hweALL2$P<1E-5),]  # which SNPs have
a p-val less than 1E-5? 289
```

```
hwe2_less10_4 <- hweALL2[which(hweALL2$P<1E-4),]  # 517 which is not
quite the same as in PLINK as above


# 2. Set minimum MAF
plink --noweb --bfile cutoff_15_clean_hwe_1E-4 --maf 0.05 --out
cutoff_15_clean3_maf
# 1,108728 SNPs failed frequency test ( MAF < 0.05 )
# 1221794 SNPs
# I think this is too stringent as it will cut of everything under
5% frequency which could be quite
# interesting for familial data.  will set to 0.01 and then
investigate what ppl using set it to

plink --noweb --bfile cutoff_15_clean_hwe_1E-4 --maf 0.01 --out
cutoff_15_clean3_maf_0.01
# 829,316 SNPs failed frequency test ( MAF < 0.01 )

plink --noweb --bfile cutoff_15_clean_hwe_1E-4 --maf 0.001 --make-
bed --out cutoff_15_clean3_maf_0.001
# 713,687 SNPs failed frequency test ( MAF < 0.001 )
# Use this cutoff.
# 1,616835 SNPs remaining


# ---- final PLINK file:  cutoff_15_clean3_maf_0.001 ----------


# missingness, hwe, het
plink --noweb --bfile cutoff_15_clean3_maf_0.001 --missing --hwe
1E-4 --hardy --het --make-bed --out cutoff_15_final
#1616835 SNPs
#12543 heterozygous haploid genotypes; set to missing
#Total genotyping rate in remaining individuals is 0.993451
plink --noweb --bfile cutoff_15_final --het --out cutoff_15_final




# create the individual (missingness vs homozygosity) and per SNP
(missingness vs hwe) plots using this

# ------- Per Individual ---------
# plot homozygosity against errors per individual
het_1.5 <- read.delim("cutoff_15_final.het",as.is=T,header=T,sep="")
imiss_1.5 <-
read.delim("cutoff_15_final.imiss",as.is=T,header=T,sep="")
snp_1.5 <-
read.delim("cutoff_15_final.lmiss",header=T,as.is=T,sep="")

png("Individual 1.5 cutoff Post QC.png")
plot(het_1.5$O.HOM./
het_1.5$N.NM.,imiss_1.5$F_MISS,ylab="missingness",xlab="Homozygosity
",
```

```
     main="Summary per individual, Post QC (51 samples 1.5
cutoff)",cex=1.5,cex.lab=1.5,cex.axis=1.5,cex.main=1.5,)
dev.off()
# compare this to pre-QC
plink --noweb --tfile cutoff_15 --het --missing --out
cutoff_15_preQC
#2391739 SNPs
#Total genotyping rate in remaining individuals is 0.991194
het_1.5pre <-
read.delim("cutoff_15_preQC.het",as.is=T,header=T,sep="")
imiss_1.5pre <-
read.delim("cutoff_15_preQC.imiss",as.is=T,header=T,sep="")
snp_1.5pre <-
read.delim("cutoff_15_preQC.lmiss",header=T,as.is=T,sep="")
png("Individual 1.5 cutoff Pre QC.png")
plot(het_1.5pre$O.HOM./het_1.5pre$N.NM.,imiss_1.5pre
$F_MISS,ylab="missingness",xlab="Homozygosity",
     main="Summary per individual, Pre QC (51 samples 1.5
cutoff)",cex=1.5,cex.lab=1.5,cex.axis=1.5,cex.main=1.5,)
dev.off()
# ----- heterozygosity ------
# plot missingness against heterozygosity
png("Individual 1.5 cutoff Post QC, heterozygosity.png")
plot(1-(het_1.5$O.HOM./
het_1.5$N.NM.),imiss_1.5$F_MISS,ylab="missingness",xlab="Heterozygos
ity",
     main="Summary per individual, Post QC,
     heterozygosity (51 samples 1.5
cutoff)",cex=1.5,cex.lab=1.5,cex.axis=1.5,cex.main=1.5,)
dev.off()
png("Individual 1.5 cutoff Pre QC, heterozygosity.png")
plot(1-(het_1.5pre$O.HOM./het_1.5pre$N.NM.),imiss_1.5pre
$F_MISS,ylab="missingness",xlab="Homozygosity",
     main="Summary per individual, Pre QC,
     heterozygosity (51 samples 1.5
cutoff)",cex=1.5,cex.lab=1.5,cex.axis=1.5,cex.main=1.5,)
dev.off()



# ------- per SNP --------
hardy_1.5post=read.delim("cutoff_15_final.hwe", header=T, as.is=T,
sep="")
hardy_1.5postb <- hardy_1.5post[seq(1,nrow(hardy_1.5post),3),]
# take out dodge ones 1614455 1614479
hardy_1.5postb <- hardy_1.5postb[-c(1614455,1614479),]
p_postb <- hardy_1.5postb$P
snp_1.5post <-snp_1.5[-c(1614455,1614479),]
#png("Missing per SNP vs HWE Post QC
zoomC.png",width=800,height=500)
#par(mar=c(6,5,4,2)+0.1)
plot(-1*log10(snp_1.5post$F_MISS),-1*log10(p_postb),ylab="-log10(HW-
pval)",xlab="-log10(missingness)",main="Missingness vs HWE p-value,
```

```
Post QC",ylim=c(-3/75,2))
abline(v=-1*log10(0.05),col=2)
text(-1*log10(0.05),-3/75,"5%",col=2)
abline(v=-1*log10(0.1),col=3)
text(-1*log10(0.1),-3/75,"10%",col=3)
abline(v=-1*log10(0.025),col=4)
text(-1*log10(0.025),-3/75,"2.5%",col=4)
lines(lowess(-1*log10(snp_1.5post$F_MISS[snp_1.5post$F_MISS!=0 & !
is.na(p_postb) & !is.na(snp_1.5post
$F_MISS)]),-1*log10(p_post[snp_1.5post$F_MISS!=0 & !is.na(p_postb)
& !is.na(snp_1.5post$F_MISS)]),f=0.1),col="red",lwd=2)

#### This still doesn't work!!!!!! -- maybe several points there
#identify(-1*log10(snp_1.5post$F_MISS),-1*log10(p_postb), n=2)
# [1] 1614455 1614479
#dev.off()

# compare this to pre-QC
plink --noweb --tfile cutoff_15 --hardy --missing --out
cutoff_15_preQC
hardy_1.5pre=read.delim("cutoff_15_preQC.hwe", header=T, as.is=T,
sep="")
p_pre <- hardy_1.5pre$P[3*(1:(length(hardy_1.5pre$P)/3))-2]
png("Missing per SNP vs HWE Pre QC zoomC.png",width=800,height=500)
par(mar=c(6,5,4,2)+0.1)
plot(-1*log10(snp_1.5pre$F_MISS),-1*log10(p_pre),ylab="-log10(HW-
pval)",xlab="-log10(missingness)",main="Missingness vs HWE p-value,
Pre QC",ylim=c(-3/75,2))
abline(v=-1*log10(0.05),col=2)
text(-1*log10(0.05),-3/75,"5%",col=2)
abline(v=-1*log10(0.1),col=3)
text(-1*log10(0.1),-3/75,"10%",col=3)
abline(v=-1*log10(0.025),col=4)
text(-1*log10(0.025),-3/75,"2.5%",col=4)
lines(lowess(-1*log10(snp_1.5pre$F_MISS[snp_1.5pre$F_MISS!=0 & !
is.na(p_pre) & !is.na(snp_1.5pre$F_MISS)]),-1*log10(p_pre[snp_1.5pre
$F_MISS!=0 & !is.na(p_pre) & !is.na(snp_1.5pre
$F_MISS)]),f=0.1),col="red",lwd=2)
dev.off()
identify(-1*log10(snp_1.5pre$F_MISS),-1*log10(p_pre), n=2)
# [1] 2386673 2386736
# get rid of these and then recheck graph
snp_1.5pre2 <-snp_1.5pre[-c(2386673,2386736),]

plot(-1*log10(snp_1.5preb$F_MISS),-1*log10(p_pre),ylab="-log10(HW-
pval)",xlab="-log10(missingness)",main="Missingness vs HWE p-value,
Pre_b QC",ylim=c(-3/75,2))
abline(v=-1*log10(0.05),col=2)
text(-1*log10(0.05),-3/75,"5%",col=2)
abline(v=-1*log10(0.1),col=3)
text(-1*log10(0.1),-3/75,"10%",col=3)
abline(v=-1*log10(0.025),col=4)
text(-1*log10(0.025),-3/75,"2.5%",col=4)
lines(lowess(-1*log10(snp_1.5preb$F_MISS[snp_1.5preb$F_MISS!=0 & !
```

```
is.na(p_pre) & !is.na(snp_1.5pre$F_MISS)]),-1*log10(p_pre[snp_1.5pre
$F_MISS!=0 & !is.na(p_pre) & !is.na(snp_1.5pre
$F_MISS)]),f=0.1),col="red",lwd=2)


save.image("myPipeline_15cutoff.RData")


##########################################################

#  cutoff 5
~/Data/Omni25/pipeline_omni
plink --noweb --file All_Chips_5_0Cutoff --transpose --recode --out
cutoff_50
nano cutoff_50.tfam
plink --noweb --tfile cutoff_50 --check-sex --out sexcheck_50
plink --noweb --tfile cutoff_50 --mind 0.1 --geno 0.1 --make-bed --
out cutoff_50_clean
plink --noweb --bfile cutoff_50_clean --maf 0.001 --make-bed --out
cutoff_50_clean_maf_0.001
plink --noweb --bfile cutoff_50_clean_maf_0.001 --missing --hwe 1E-4
--hardy --het --make-bed --out cutoff_50_final
plink --noweb --bfile cutoff_50_final --het --out cutoff_50_final

# plot homozygosity against errors per individual
R
setwd("~/Data/Omni25/pipeline_omni")
het_50 <- read.delim("cutoff_50_final.het",as.is=T,header=T,sep="")
imiss_50 <-
read.delim("cutoff_50_final.imiss",as.is=T,header=T,sep="")
snp_50 <-
read.delim("cutoff_50_final.lmiss",header=T,as.is=T,sep="")

png("Individual 5.0 cutoff Post QC.png")
plot(het_50$O.HOM./
het_50$N.NM.,imiss_50$F_MISS,ylab="missingness",xlab="Homozygosity",
     main="Summary per individual, Post QC (51 samples 5.0
cutoff)",cex=1.5,cex.lab=1.5,cex.axis=1.5,cex.main=1.5,)
dev.off()
```

```
## Chapter.3 Appendix_1 -- perform normalisation ##



## Overview ##
_____________________________________________________________________
_________________
# first 5 steps performed in ch.2 as part of QC protocol
## 1. Read in data
## 2. Remove failed samples
## 3. Background correction & Control normalisation
## 4. Convert from RGset to MethylSet
## 5. Remove sex chr and normalise separately then recombine.  This
is tricky for Methylumi objects as annotation seems impossible

# 6. Normalisation: swan, QN, BMIQ, Dasen
# 7. Remove failed probes based on detP- nonsignificant signal
compared to background 0.05 threshold
# 8. Batch effect correction: limma + RUV, Combat
_____________________________________________________________________
_________________


#####################################################################
###############
# 6. Normalisation:   Compare RAW to StratQN, FunNorm, SWAN, QN,
BMIQ, Dasen, RUV #
#####################################################################
###############

# use Autosomes object as need to normalise sex chr separately,
still need to put RGsetE2 argument but make the mset the Autosomes
object and then do the same for both sex chr

### 1. RAW ###
Raw_preprocess
Autosomes
Male_chr
Female_chr

autosomes_RGsetE2 <- RGsetE2[chrs!="chrX" & chrs !="chrY"]
dim(autosomes_RGsetE2 )


###  2. Stratified Quantile Normalisation ###
#Stratified quantile normalization for Illumina amethylation arrays.
This function implements stratified quantile normalization
preprocessing for Illumina methylation microarrays. Probes are
stratified by region (CpG island, shore, etc.) This function
implements stratified quantile normalization preprocessing for
Illumina methylation microarrays. If removeBadSamples is TRUE we
calculate the median Meth and median Unmeth sig- nal for each
sample, and remove those samples where their average falls below
badSampleCutoff. The normalization procedure is applied to the Meth
```

and Unmeth intensities separately. The distri- bution of type I and type II signals is forced to be the same by first quantile normalizing the type II probes across samples and then interpolating a reference distribution to which we normalize the type I probes. Since probe types and probe regions are confounded and we know that DNAm dis- tributions vary across regions we stratify the probes by region before applying this interpolation. For the probes on the X and Y chromosomes we normalize males and females separately using the gender information provided in the sex argument. If gender is unspecified (NULL), a guess is made using by the getSex function using copy number information. Background correction is not used, but very small intensities close to zero are thresholded using the fixMethOutlier. Note that this algorithm relies on the assumptions necessary for quantile normalization to be applicable and thus is not recommended for cases where global changes are expected such as in cancer-normal comparisons.
Note that this normalization procedure is essentially similar to one previously presented (Touleimat and Tost, 2012), but has been independently re-implemented due to the present lack of a released, supported version.

```
library(minfi)
StratQN <- preprocessQuantile(Autosomes, fixOutliers=TRUE,
removeBadSamples=FALSE, quantileNormalize=TRUE, stratified=TRUE, ,
mergeManifest=TRUE, sex=Autosomes@phenoData$Sex, verbose=TRUE)
# can't use RGset as get an error saying subset out of bounds
# dont actually need to use the autosome object as the function
normalised sex chr separately when sex is provided
type_autosomes <- data.frame(row.names(StratQN),
getProbeType(StratQN))
colnames(type_autosomes) <- c("Name", "Type")
StratQN_betas <- getBeta(StratQN)

StratQN_all <- preprocessQuantile(Raw_preprocess, fixOutliers=TRUE,
removeBadSamples=FALSE, quantileNormalize=TRUE, stratified=TRUE, ,
mergeManifest=TRUE, sex= Raw_preprocess@phenoData$Sex, verbose=TRUE)



### 3. Functional Normalisation ###
library(minfi)
FunNorm <- preprocessFunnorm(RGsetE2, nPCs=2, sex=RGsetE2@phenoData
$Sex, bgCorr=TRUE, dyeCorr=TRUE, verbose=TRUE)
FunNorm_betas <- getBeta(FunNorm)

#FunNorm_autosomes <- preprocessFunnorm(autosomes_RGsetE2, nPCs=2,
sex=autosomes_RGsetE2@phenoData$Sex, bgCorr=TRUE, dyeCorr=TRUE)
# Error in getGreen(object)[IRed$AddressA, ] : subscript out of
bounds
# Also can't use Autosomes as needs to be RGset object not methylset

probeTypes_FunNorm <- data.frame(row.names(FunNorm),
getProbeType(FunNorm))
colnames(probeTypes_FunNorm) <- c("Name", "Type")
```

```
par(mfrow=c(1,2))
plotBetasByType(getBeta(FunNorm)[,7], probeTypes=probeTypes_FunNorm,
main="FunNorm by probe")
plotBetasByType(swan2[,7], probeTypes=probeTypes_FunNorm,
main="RGsetE2 by probe")
# These look quite different, there is a greater skew towards
unmethylated for the normalised data whereas the RGset has much more
methylated



### 4. SWAN ###
swan=preprocessSWAN(RGsetE2, mSet=Raw_preprocess)
# will perform preprocessRaw on the RGset if NULL for mset, this
looks diff between my Raw_preprocess which I've done background
correction on and the default, don't use default
#swan2=preprocessSWAN(RGsetE2) #default
#par(mfrow=c(1,2))
#plotBetasByType(swan[,7], main="SWAN by probe, my Raw")
#plotBetasByType(swan2[,7], main="SWAN by probe, default")

swan_autosomes <- preprocessSWAN(RGsetE2, mSet=Autosomes)
swan_maleChr <- preprocessSWAN(RGsetE2, mSet=Male_chr)
swan_femaleChr <- preprocessSWAN(RGsetE2, mSet=Female_chr)

par(mfrow=c(1,3))
plotBetasByType(swan_autosomes[,7], main="SWAN by probe, autosomes")
plotBetasByType(swan_maleChr[,7], main="SWAN by probe, male chr")
plotBetasByType(swan_femaleChr[,7], main="SWAN by probe, female
chr")
# female chr look pretty odd, guessing this is a reflection of X
inactivation



### 5. QN ###
library(methylumi)
library(IlluminaHumanMethylation450kanno.ilmn12.hg19)
library(FDb.InfiniumMethylation.hg19)
pdat=read.csv("/Users/ecazaly/Desktop/R Dec14/Raw IDATs/Methylumi/
Methylumi_pdat2.csv")
MethylumiSet=methylumIDAT(barcodes = pdat$barcodes, pdat = pdat,
parallel = F, n = F, n.sd=F, oob = T, idatPath=file.path("/Users/
ecazaly/Desktop/R Dec14/Raw IDATs/Methylumi"))
bad_methylumi=c(9,17,22,24,27,29,40,42)
MethylumiSet2=MethylumiSet[,-c(bad_methylumi)]
## Background and colour correction ##
library(lumi)
methylumi_prepro <-
normalizeMethyLumiSet(methylumi.bgcorr(MethylumiSet2))
# For HumanMethylation450 data, the function delegates to
normalizeViaControls() the task of scaling red and green intensities
against a reference array (chip) which defaults to the first chip in
a set. The code to do this is based on code from the 'minfi' package
```

```
and uses the built-in normalization controls to scale the #channels
of the samples, so that a consistent degree of dye bias is
maintained for Infinium II probes across an experiment or set of
experiments.
### QN ###
lumiQN_prepro=normalizeMethylation.quantile(methylumi_prepro)
lumiQNbetas=betas(lumiQN_prepro)
type_methylumi <- data.frame(row.names(lumiQN_prepro),
lumiQN_prepro@featureData@data$DESIGN)
colnames(type_methylumi) <- c("Name", "Type")
plotBetasByType(lumiQNbetas[,1], probeTypes=type_methylumi,
main="lumiQNbetas by probe")



### 6. BMIQ ###
library(wateRmelon)
library(RPMM)
# BMIQ_autosomes=BMIQ(getBeta(Autosomes), probeTypes_autosomes) #
this doesn't work, looks like you need all the probes,
Raw_preprocess and RGset also don't either
BMIQ=BMIQ(methylumi_prepro) # just have to do it with all
chromosomes
BMIQ_betas <- betas(BMIQ)
type_BMIQ <- data.frame(row.names(BMIQ), BMIQ@featureData@data
$DESIGN)
colnames(type_BMIQ) <- c("Name", "Type")



### 7. Dasen ###
Dasen=as.matrix(dasen(methylated(Raw_preprocess),unmethylated(Raw_pr
eprocess),onetwo= getProbeType(Raw_preprocess), fudge=100))
probeTypes_Dasen <- data.frame(row.names(Raw_preprocess),
getProbeType(Raw_preprocess))
        -       colnames(probeTypes_Dasen) <- c("Name", "Type")

Dasen_autosomes=as.matrix(dasen(methylated(Autosomes),unmethylated(A
utosomes),onetwo=getProbeType(Autosomes), fudge=100))

probeTypes_autosomes2 <- data.frame(row.names(Autosomes),
getProbeType(Autosomes))
colnames(probeTypes_autosomes2) <- c("Name", "Type")
plotBetasByType(Dasen[,1], probeTypes= probeTypes_autosomes2,
main="Dasen by probe, autosomes")

probeTypes_male <- getProbeType(Male_chr)
Dasen_male=as.matrix(dasen(methylated(Male_chr),unmethylated(Male_ch
r),onetwo= probeTypes_male, fudge=100))
probeTypes_autosomes2_male <- data.frame(row.names(Male_chr),
getProbeType(Male_chr))
colnames(probeTypes_autosomes2_male) <- c("Name", "Type")
plotBetasByType(Dasen_male[,1], probeTypes=
probeTypes_autosomes2_male, main="Dasen by probe, male chr")
```

```
# the plot works for Dasen but not male/female chr for some reason
probeTypes_female <- getProbeType(Female_chr)
Dasen_female=as.matrix(dasen(methylated(Female_chr),unmethylated(Fem
ale_chr),onetwo= probeTypes_female, fudge=100))
probeTypes_autosomes2_female <- data.frame(row.names(Female_chr),
getProbeType(Female_chr))
colnames(probeTypes_autosomes2_female) <- c("Name", "Type")
plotBetasByType(Dasen_female[,1], probeTypes=
probeTypes_autosomes2_female, main="Dasen by probe, female chr")



### 8. RUV ###

library(minfi)
library(methylumi)
#library(RUVnormalize)
source("naiveRandRUV.R")
source("missMethylRUVFunctions.R")

mValues_autosomes = getMvals(Autosomes,RGsetE2)
ctl_autosomes <- !(rownames(mValues_autosomes) %in%
featureNames(Autosomes))  ## create logical vector marking NCPs, 613
probes from 479078- 479690
mValues_raw <- getMvals(Raw_preprocess, RGsetE2)
ctl_raw <- !(rownames(mValues_raw) %in%
featureNames(Raw_preprocess))

result_autosomes= naiveRandRuv(t(mValues_autosomes), ctl_autosomes,
nuCoeff=1e-3, k=ceiling(ncol(mValues_autosomes)/4)) ## perform
normalization
RUV_autosomes=t(result_autosomes)
BetaRUV_autosomes=ilogit2(RUV_autosomes)

densityPlot(BetaRUV_autosomes,sampGroups=Autosomes@phenoData$Batch,
pal = rainbow(3), xlim=c(-.1,1.2),ylim=c(0,20), xlab="Beta",
main="Density plot RUV data, Autosomes")
type=data.frame(cbind(rownames(Autosomes),getProbeType(Autosomes)))
colnames(type)=c("Name","Type")

type_male=data.frame(cbind(rownames(Male_chr),getProbeType(Male_chr)
))
colnames(type_male)=c("Name","Type")
type_female=data.frame(cbind(rownames(Female_chr),getProbeType(Femal
e_chr)))
colnames(type_female)=c("Name","Type")

plotBetasByType(BetaRUV_autosomes[,1], type , main="Autosomes
BetaRUV by type", cex.legend=.8, legendPos="topright")
mdsPlot(BetaRUV_autosomes, numPositions=1000,
sampNames=Autosomes@phenoData$Sample_Name,
sampGroups=Autosomes@phenoData$Batch,
        xlim=c(-14,16), ylim=c(-7,5),legendNCol=5, legendPos =
"bottom", main="MDS- RUV Autosomes")
```

```
# change variables when generating result and see if this affects
the density plot
result_autosomes_by2= naiveRandRuv(t(mValues_autosomes),
ctl_autosomes, nuCoeff=1e-3, k=ceiling(ncol(mValues_autosomes)/2))
## perform normalization
result_autosomes_by6= naiveRandRuv(t(mValues_autosomes),
ctl_autosomes, nuCoeff=1e-3, k=ceiling(ncol(mValues_autosomes)/6))
## perform normalization
result_autosomes_by8= naiveRandRuv(t(mValues_autosomes),
ctl_autosomes, nuCoeff=1e-3, k=ceiling(ncol(mValues_autosomes)/8))
## perform normalization
RUV_autosomes_by2=t(result_autosomes_by2)
BetaRUV_autosomes_by2=ilogit2(RUV_autosomes_by2)
RUV_autosomes_by6=t(result_autosomes_by6)
BetaRUV_autosomes_by6=ilogit2(RUV_autosomes_by6)
RUV_autosomes_by8=t(result_autosomes_by8)
BetaRUV_autosomes_by8=ilogit2(RUV_autosomes_by8)
par(mfrow=c(1,3))
densityPlot(BetaRUV_autosomes_by2,sampGroups=Autosomes@phenoData
$Batch, pal = rainbow(3), xlim=c(-.1,1.2),ylim=c(0,20), xlab="Beta",
main="Density plot RUV data by2, Autosomes")
densityPlot(BetaRUV_autosomes_by6,sampGroups=Autosomes@phenoData
$Batch, pal = rainbow(3), xlim=c(-.1,1.2),ylim=c(0,20), xlab="Beta",
main="Density plot RUV data by6, Autosomes")
densityPlot(BetaRUV_autosomes_by8,sampGroups=Autosomes@phenoData
$Batch, pal = rainbow(3), xlim=c(-.1,1.2),ylim=c(0,20), xlab="Beta",
main="Density plot RUV data by8, Autosomes")
# changes the height of the graph

result_autosomes_nuCoeff_1e5= naiveRandRuv(t(mValues_autosomes),
ctl_autosomes, nuCoeff=1e-5, k=ceiling(ncol(mValues_autosomes)/4))
## perform normalization
result_autosomes_nuCoeff_1e1= naiveRandRuv(t(mValues_autosomes),
ctl_autosomes, nuCoeff=1e-1, k=ceiling(ncol(mValues_autosomes)/4))
## perform normalization
result_autosomes_nuCoeff_1ePos5= naiveRandRuv(t(mValues_autosomes),
ctl_autosomes, nuCoeff=1e5, k=ceiling(ncol(mValues_autosomes)/4)) ##
perform normalization
RUV_autosomes_nuCoeff_1e5=t(result_autosomes_nuCoeff_1e5)
BetaRUV_autosomes_nuCoeff_1e5=ilogit2(RUV_autosomes_nuCoeff_1e5)
RUV_autosomes_nuCoeff_1e1=t(result_autosomes_nuCoeff_1e1)
BetaRUV_autosomes_nuCoeff_1e1=ilogit2(RUV_autosomes_nuCoeff_1e1)
RUV_autosomes_nuCoeff_1ePos5=t(result_autosomes_nuCoeff_1ePos5)
BetaRUV_autosomes_nuCoeff_1ePos5=ilogit2(RUV_autosomes_nuCoeff_1ePos
5)
par(mfrow=c(1,3))
densityPlot(BetaRUV_autosomes_nuCoeff_1e5,sampGroups=Autosomes@pheno
Data$Batch, pal = rainbow(3), xlim=c(-.1,1.2),ylim=c(0,20),
xlab="Beta", main="Density plot RUV data nuCoeff 1e-5, Autosomes")
densityPlot(BetaRUV_autosomes_nuCoeff_1e1,sampGroups=Autosomes@pheno
Data$Batch, pal = rainbow(3), xlim=c(-.1,1.2),ylim=c(0,20),
xlab="Beta", main="Density plot RUV data nuCoeff 1e-1, Autosomes")
densityPlot(BetaRUV_autosomes_nuCoeff_1ePos5,sampGroups=Autosomes@ph
```

```
enoData$Batch, pal = rainbow(3), xlim=c(-.1,1.2),ylim=c(0,20),
xlab="Beta", main="Density plot RUV data nuCoeff 1e +5, Autosomes")
# this one is starting to look better.  Back to being biomdal
distribution

# from the above looks like nuCoeff of ~5 and dividing by ~6 is best
result_autosomes_optimal= naiveRandRuv(t(mValues_autosomes),
ctl_autosomes, nuCoeff=1e5, k=ceiling(ncol(mValues_autosomes)/6)) ##
perform normalization
RUV_autosomes_optimal=t(result_autosomes_optimal)
BetaRUV_autosomes_optimal=ilogit2(RUV_autosomes_optimal)
densityPlot(BetaRUV_autosomes_optimal,sampGroups=Autosomes@phenoData
$Batch, pal = rainbow(3), xlim=c(-.1,1.2),ylim=c(0,3.5),
xlab="Beta", main="Density plot RUV optimal, Autosomes")

#compare:
par(mfrow=c(1,3))
densityPlot(getBeta(Autosomes),sampGroups=Autosomes@phenoData$Batch,
pal = rainbow(3), xlim=c(-.1,1.2),ylim=c(0,3.5), xlab="Beta",
main="Density plot Preprocessed Autosomes")
densityPlot(BetaRUV_autosomes,sampGroups=Autosomes@phenoData$Batch,
pal = rainbow(3), xlim=c(-.1,1.2),ylim=c(0,20), xlab="Beta",
main="Density plot RUV Autosomes")
densityPlot(BetaRUV_autosomes_optimal,sampGroups=Autosomes@phenoData
$Batch, pal = rainbow(3), xlim=c(-.1,1.2),ylim=c(0,3.5),
xlab="Beta", main="Density plot RUV optimal, Autosomes")

mdsPlot(getBeta(Autosomes), numPositions=1000,
sampNames=Autosomes@phenoData$Sample_Name,
sampGroups=Autosomes@phenoData$Batch,
        xlim=c(-16,16), ylim=c(-7,5),legendNCol=5, legendPos =
"bottom", main="MDS- Preprocessed Autosomes")
mdsPlot(BetaRUV_autosomes, numPositions=1000,
sampNames=Autosomes@phenoData$Sample_Name,
sampGroups=Autosomes@phenoData$Batch,
        xlim=c(-16,16), ylim=c(-7,5),legendNCol=5, legendPos =
"bottom", main="MDS- RUV Autosomes")
mdsPlot(BetaRUV_autosomes_optimal, numPositions=1000,
sampNames=Autosomes@phenoData$Sample_Name,
sampGroups=Autosomes@phenoData$Batch,
        xlim=c(-16,16), ylim=c(-7,5),legendNCol=5, legendPos =
"bottom", main="MDS- RUV Autosomes_optimal")
# the raw and optimal look pretty similar and there is still a batch
effect after RUV when I adjust the variables

# Male and female chr
mValues_male = getMvals(Male_chr,RGsetE2)
ctl_male <- !(rownames(mValues_male) %in% featureNames(Male_chr))
result_male= naiveRandRuv(t(mValues_male), ctl_male, nuCoeff=1e-3,
k=ceiling(ncol(mValues_male/4)))
RUV_male=t(result_male)
BetaRUV_male=ilogit2(RUV_male)

mValues_female <- getMvals(Female_chr, RGsetE2)
```

```
ctl_female <- !(rownames(mValues_female) %in%
featureNames(Female_chr))
result_female= naiveRandRuv(t(mValues_female), ctl_female,
nuCoeff=1e-3, k=ceiling(ncol(mValues_female/4)))
RUV_female=t(result_female)
BetaRUV_female=ilogit2(RUV_female)

# Male and female OPTIMAL
result_male_optimal= naiveRandRuv(t(mValues_male), ctl_male,
nuCoeff=1e5, k=ceiling(ncol(mValues_male)/6))
RUV_male_optimal=t(result_male_optimal)
BetaRUV_male_optimal=ilogit2(RUV_male_optimal)

result_female_optimal= naiveRandRuv(t(mValues_female), ctl_female,
nuCoeff=1e5, k=ceiling(ncol(mValues_female)/6))
RUV_female_optimal=t(result_female_optimal)
BetaRUV_female_optimal=ilogit2(RUV_female_optimal)



### 7. Remove failed probe sites based on detection P value ###
# compared to background.  I think 0.01 is too stringent, use 0.05
which is the default for wateRmelon.
####
# Creates an RGset, dan't use this method on MethylSet, needs to be
RGset
detP <- detectionP(RGsetE2_control_norm)  # from minfi
failed_detP=detP>0.05
failed_sites_0.05_1=which(rowMeans(failed_detP)>0.001) # Which
positions failed in >1% of samples?
Raw_minfi_0.05_1=RGsetE2_control_norm[-(failed_sites_0.05_1),]
dim(Raw_minfi_0.05_1)
#Features  Samples
#615237       52
dim(RGsetE2_control_norm)-dim(Raw_minfi_0.05_1)
# Features  Samples
# 7162        0   # this is the same as when you use the wateRmelon
method below :)
# But there are less sites in Raw_filter because it has been
converted to a methylset and this removes probes
dim(Raw_minfi_0.05_1)-dim(Raw_filter)  # Features  Samples 2450
0

## OR:  Use pfilter method wateRmelon, removes 2450 probes, still
RGset
# this is still a lot less than removed when using preprocess
(136,887)

Raw_filter  <- pfilter(mn=RGsetE2_control_norm,pn=detP)    # Can
change threshold from default of 0.05 to 0.01 with additional
arguments
# 0 samples having 1 % of sites with a detection p-value greater
than 0.05 were removed
```

```
# 719 sites were removed as beadcount <3 in 5 % of samples
# 7162 sites having 1 % of samples with a detection p-value greater
than 0.05 were removed
dim(Raw_filter)
#  Features  Samples
#  612787       52
densityPlot(Raw_filter, sampGroups=Raw_filter@phenoData$Batch,
pal=rainbow(3), xlim=c(-.1,1.2), ylim=c(0,3.6), xlab="Beta",
main="Raw- pfilter")
mdsPlot(Raw_filter, numPositions=1000,
sampNames=Raw_filter@phenoData$Sample_Name,
sampGroups=Raw_filter@phenoData$Batch,
        xlim=c(-8.5,5), ylim=c(-5,8),legendNCol=5, legendPos =
"bottom", main="MDS Raw_filter")
# can't plot:  Error in getGreen(rgSet)[getProbeInfo(rgSet, type =
"II")$AddressA, ] : subscript out of bounds

# methylumi #
require(wateRmelon)
detP_methylumi <- pval.detect(MethylumiSet2)
methylumiFILTER=pfilter(mn=MethylumiSet2,pn=detP_methylumi,
pnthresh=0.05)
# 0 samples having 1 % of sites with a detection p-value greater
than 0.05 were removed
# 1437 sites were removed as beadcount <3 in 5 % of samples
# 5604 sites having 1 % of samples with a detection p-value greater
than 0.05 were removed

detP_methylumi_prepro=pval.detect(methylumi_prepro)
methylumiFILTER_preprop=pfilter(mn=methylumi_prepro,pn=detP_methylum
i_prepro, pnthresh=0.05)
#this removes the same number of sites, shouldnt it remove less
since there is background correction?
# pval.detect does sfa, it just generates one value of 0.05 for all
the data



###  8. batch correction; Combat, RUV + limma, ISVA, CpGassoc ###
library(minfi)
library(methylumi)
#library(RUVnormalize)
source("naiveRandRUV.R")
source("missMethylRUVFunctions.R")

### Combat ###
library(sva)
ComBat_autosomes=ComBat(dat=getBeta(Autosomes),
batch=Autosomes@phenoData$Batch,mod=1, par.prior=TRUE,
prior.plots=FALSE)
#Found 3 batches Found 0  categorical covariate(s) Found 908 Missing
Data Values Standardizing Data across genes Fitting L/S model and
finding priors Finding parametric adjustments Adjusting the Data
```

```
### limma correction then RUV ###
#################################
library(limma)
mValues2 = getMvals(Autosomes,RGsetE2)  ## calculate M-values from
intensity data, include negative control probes (NCPs)
## create design matrix, this can include additional covariates
(i.e. it is X and Z combined)
batch <- factor(Autosomes@phenoData$Batch)
design <- model.matrix(~batch)
coef= colnames(design)[2]
fit = lmFit(mValues2, design)
fit = eBayes(fit)
result2 = topTable(fit,coef=coef,num=Inf)
ctl2 <- rownames(mValues2) %in% rownames(result2)
[ceiling(nrow(result2)*0.5):nrow(result2)]  ## select empirical
controls
results = naiveRandRuv(t(mValues2), ctl2, nuCoeff=1e-3,
k=ceiling(ncol(mValues2)/4)) ## perform normalization
RUV_limma=t(results)
BetaRUV_limma=ilogit2(RUV_limma)

par(mfrow=c(1,2))
densityPlot(BetaRUV_limma,sampGroups=Autosomes@phenoData$Batch, pal
= rainbow(3), xlim=c(-.1,1.2),ylim=c(0,20), xlab="Beta",
main="Density plot BetaRUV_limma, Autosomes")
mdsPlot(BetaRUV_limma, numPositions=1000,
sampNames=Raw_autosomes@phenoData$Sample_Name,
sampGroups=Raw_autosomes@phenoData$Batch,
        xlim=c(-16,16), ylim=c(-7,5),legendNCol=5, legendPos =
"bottom", main="MDS- BetaRUV_limma")
# use optimal settings as above
results_optimal = naiveRandRuv(t(mValues2), ctl2, nuCoeff=1e5,
k=ceiling(ncol(mValues2)/6)) ## perform normalization
RUV_limma_optimal=t(results_optimal)
BetaRUV_limma_optimal=ilogit2(RUV_limma_optimal)

par(mfrow=c(1,2))
densityPlot(BetaRUV_limma_optimal,sampGroups=Raw_autosomes@phenoData
$Batch, pal = rainbow(3), xlim=c(-.1,1.2),ylim=c(0,3.5),
xlab="Beta", main="Density plot BetaRUV_limma_optimal, Autosomes")
mdsPlot(BetaRUV_limma_optimal, numPositions=1000,
sampNames=Raw_autosomes@phenoData$Sample_Name,
sampGroups=Raw_autosomes@phenoData$Batch,
        xlim=c(-16,16), ylim=c(-7,5),legendNCol=5, legendPos =
"bottom", main="MDS- BetaRUV_limma_optimal")
# this looks identical to RUV without limma


### ISVA ###
library(isva)
#Independent Surrogate Variable Analysis is an algorithm for feature
selection in the presence of potential confounding factors,
specially designed for the analysis of large-scale high-dimensional
```

quantitative genomic data (e.g microarrays). It uses Independent Component Analysis (ICA) to model the confounding factors as independent surrogate variables (ISVs). These ISVs are included as covariates in a multivariate regression model to subsequently identify features that correlate with a phenotype of interest independently of these confounders. The ICA implementation used is that of the fastICA R-package.

```
# try NULL and 3 and 5 for the batches for potential confounders
ISVAdata <-matrix(getBeta(Autosomes), nrow=nrow(Autosomes),
ncol=ncol(Autosomes))
rownames(ISVAdata) <- rownames(Autosomes)
colnames(ISVAdata) <- colnames(Autosomes)
pheno <- seq(from=1, to=1, length.out=52)
cf <-matrix(Autosomes$Batch)
rownames(cf) <-colnames(Autosomes)
colnames(cf) <-"Batch"
var <- Autosomes$Affected
var <- ifelse(var =="A", 1, 2)
isva<- DoISVA(data.m=ISVAdata, pheno.v=var, cf.m=NULL, pvthCF=0.01,
th=0.05, ncomp=NULL)
# no matterv what I do :
#Error in lm.fit(x, y, offset = offset, singular.ok =
singular.ok, ...) : 0 (non-NA) cases

isvaFn(data.m=ISVAdata, pheno.v=pheno, ncomp=NULL )
#same error as above


### CpGassoc ###
#cpg.assoc is designed to test for association between an
independent variable and methylation at a number of CpG sites, with
the option to include additional covariates and factors. cpg.assoc
assesses sig- nificance with the Holm (step-down Bonferroni) and FDR
methods.
library(CpGassoc)

cpgAss <- cpg.assoc(beta.val=ISVAdata, indep=Autosomes$Batch,
covariates= NULL, logit.transform=TRUE, chip.id=Autosomes$Slide,
random=FALSE)
#use it to check if there is a batch effect still present after norm
etc???

#Warning messages:
1: In rm(non.m.beta, sser, ssef, beta0, r.ressq) : object 'sser' not
found
2: In rm(non.m.beta, sser, ssef, beta0, r.ressq) : object 'ssef' not
found
3: In rm(non.m.beta, sser, ssef, beta0, r.ressq) :
  object 'beta0' not found
4: In rm(non.m.beta, sser, ssef, beta0, r.ressq) :
  object 'r.ressq' not found
```

```
### Streamlined processing ###
# ChAMP -- seems to be able to do practically everything and has an
inbuilt all in one pipeline command 'champ.process'
library(ChAMP)
champ <- champ.process(fromIDAT=TRUE, directory="/Users/ecazaly/
Desktop/R Dec14/Raw IDATs/IDATs_passed", methValue="B",
filterDetP=TRUE, detPcut=0.05, filterXY=TRUE, QCimages=TRUE,
filterBeads=TRUE, beadCutoff=0.05, batchCorrect=TRUE, runSVD=TRUE,
norm="SWAN", adjust.method="BH", runDMR=FALSE, runCNA=FALSE,
plotBMIQ=TRUE)
# this removes practically all the proibes because of the failed
samples.  Run only the samples that have previously passed QC

bad=c(9,17,22,24,27,29,40,42)
densityPlot(RGsetE[,bad],sampGroups=RGsetE[,bad]@phenoData$Batch,
pal = rainbow(3), xlim=c(-.1,1.2),ylim=c(0,3.6), xlab="Beta",
main="RGsetE bad")
colnames(RGsetE[,bad])
[1] "8795207059_R05C01" "8784225156_R03C01" "8784225156_R05C02"
[4] "8784225156_R06C02" "5975827011_R03C01" "5975827011_R05C01"
[7] "9221197159_R02C02" "9221197159_R03C02"
# these have been removed form the directory "IDATs_passed" used
above to call the champ function
# won't let me run combat with the batches as Sample_Group
# Combat failed...Your slides may be confounded with batch or with
each other. Analysis will proceed without batch correction
# and the normalisation is pretty shit, density plots look worse
than raw- ie. more prominent batches

# compare norm methods, have XY filter included
champ_data <- champ.load(directory="/Users/ecazaly/Desktop/R Dec14/
Raw IDATs/IDATs_passed", resultsDir="/Users/ecazaly/Desktop/R Dec14/
ChampLoad", methValue="B", filterXY=TRUE, QCimages=TRUE,
filterDetP=TRUE, detPcut=0.05, filterBeads=TRUE, beadCutoff=0.05,
filterNoCG=FALSE)
# creates a list of 6 objects which include: 1. minfi MethylSet
object, 2. minfi RGset, 3.pd, 4. intensity, 5. beta, 6. detP
# removes the same DetP failed probes as other methods: Filtering
probes with a detection p-value above 0.05 in more than one sample
has removed 6740 probes from the analysis. If a large number of
probes have been removed, ChAMP suggests you look at the
failedSample.txt file to identify potentially bad samples. Filtering
probes with a beadcount <3 in at least 5% of samples, has removed
478 from the analysis. Zeros in your dataset have been replaced with
0.000001
# The analysis will proceed with 467448 probes and 52 samples.
Takes out 10846 sex chr: 478294-467448

# it also removed sex chr
champ_Annotated<- getAnnotation(champ_data$mset)
which(champ_Annotated@listData$chr=="chrX")  #integer(0)
```

```
champ_data_withXY <- champ.load(directory="/Users/ecazaly/Desktop/R
Dec14/Raw IDATs/IDATs_passed", resultsDir="/Users/ecazaly/Desktop/R
Dec14/ChampLoad_withXY", methValue="B", filterXY=FALSE,
QCimages=TRUE, filterDetP=TRUE, detPcut=0.05, filterBeads=TRUE,
beadCutoff=0.05, filterNoCG=FALSE)
#The analysis will proceed with 478294 probes and 52 samples.


## Norm methods ##

champ_preprocess <- preprocessRaw(champ_data$rgSet)

champ_StratQN <-preprocessQuantile(champ_data$mset,
fixOutliers=TRUE, removeBadSamples=FALSE, quantileNormalize=TRUE,
stratified=TRUE, , mergeManifest=TRUE, sex= champ_data
$mset@phenoData@data$Sex, verbose=TRUE)

champ_FunNorm <- preprocessFunnorm(champ_data$rgSet, nPCs=2,
sex=champ_data$mset@phenoData@data$Sex, bgCorr=TRUE, dyeCorr=TRUE,
verbose=TRUE)

champ_SWAN <-preprocessSWAN(champ_data$rgSet, mSet=champ_data$mset)

champ_Dasen<- as.matrix(dasen(methylated(champ_data
$mset),unmethylated(champ_data$mset),onetwo= getProbeType(champ_data
$mset), fudge=100))


mValues_champ = getMvals(champ_data$mset,champ_data$rgSet)
ctl_champ <- !(rownames(mValues_champ) %in%
featureNames(champ_data))
result_champ= naiveRandRuv(t(mValues_champ), ctl_champ,
nuCoeff=1e-3, k=ceiling(ncol(mValues_champ)/4))
champ_RUV <-t(result_champ)

result_champ_optimal= naiveRandRuv(t(mValues_champ), ctl_champ,
nuCoeff=1e5, k=ceiling(ncol(mValues_champ)/6))
RUV_champ_optimal=t(result_champ_optimal)
BetaRUV_champ_optimal=ilogit2(RUV_champ_optimal)

champ_QN <-
normalizeMethylation.quantile(as.matrix(methylated(champ_data
$mset),unmethylated(champ_data$mset)))  #... doesn't work:Error in
#normalizeMethylation.quantile(champ_data$rgSet) : The input should
include 'methylated' and 'unmethylated' elements in the assayData
slot!

design.v <- getProbeType(champ_data$mset)
design.v <- ifelse(design.v=="I", "1","2")
champ_BMIQ <- BMIQ(getBeta(champ_data$mset), design.v=design.v2))
# looks the same if I poerform the norm on dat initially leaving in
the Xy but filtering during norm

# From minfi; Noob is a background correction method with dye-bias
```

```
normalization for the Illumina Infinium HumanMethylation450
platform.
champ_Noob <- preprocessNoob(champ_data$rgSet,offset=15,
dyeCorr=TRUE, verbose=TRUE)

densityPlot(champ_Noob,sampGroups= champ_Noob@phenoData$Batch, pal =
rainbow(3), xlim=c(-.1,1),ylim=c(0,4), xlab="Beta", main="Density
plot champ_Noob data")
mdsPlot(champ_Noob, numPositions=1000, sampNames=
champ_Noob@phenoData$Sample_Name, sampGroups= champ_Noob@phenoData
$Batch, xlim=c(-16,16), ylim=c(-7,5),legendNCol=5, legendPos =
"bottom", main="MDS- champ_Noob")
```

```
## Chapter.3 Appendix_2:  test normalisation methods ##



## Performance Metrics ##

#1.  Density plots; to examine QC samples, batch effects and also
type I vs II distribution z1
    plotBetasByType

#2.     MDS / PCA plots:
# a.possibly with wilcoxon test to check for associations between
the first few PCs/dimensions and batch variable princomp= function
# b.    ANOVA to check for the proportion of CpGs associated with
batches?

#3.     Unsupervised hierarchical clustering

#4.     Replicate samples
# a.    visually examining density and MDS plots
# b.    Mean absolute difference in M values between replicates-
may need help with an algorithm for this if I can't find something
in a package or supp material

#5.     Diagnostic methods in the wateRmelon package; iDMRs, XCI,
65snps - standard error type measurement for each (DMRSE, 1-AUC,
GCOSE)

#6. Positive Controls



##########################################
##  1. Density plots & plotBetasByType  ##
##########################################

# Raw
par(mfrow=c(1,4))
densityPlot(Raw_preprocess,sampGroups=Raw_preprocess@phenoData
$Batch, pal = rainbow(3), xlim=c(-.1,1.2),ylim=c(0,3.6),
xlab="Beta", main="Raw_preprocess")
densityPlot(Autosomes,sampGroups=Autosomes@phenoData$Batch, pal =
rainbow(3), xlim=c(-.1,1.2),ylim=c(0,3.6), xlab="Beta",
main="Autosomes_preprocess")
densityPlot(Male_chr,sampGroups= Male_chr@phenoData$Batch, pal =
rainbow(3), xlim=c(-.1,1.2),ylim=c(0,3.6), xlab="Beta",
main="Male_chr")
densityPlot(Female_chr,sampGroups= Female_chr@phenoData$Batch, pal =
rainbow(3), xlim=c(-.1,1.2),ylim=c(0,3.6), xlab="Beta",
main="Female_chr")

par(mfrow=c(1,4))
plotBetasByType(Raw_preprocess[,7], main="Raw by probe, all")
plotBetasByType(Autosomes[,7], main="Raw by probe, Autosomes")
```

```
plotBetasByType(Male_chr[,7], main="Raw by probe, Male_chr")
plotBetasByType(Female_chr[,7], main="Raw by probe, Female_chr")

# Stratified QN: StratQN
par(mfrow=c(1,4))
densityPlot(getBeta(StratQN),sampGroups=Autosomes@phenoData$Batch,
pal = rainbow(3), xlim=c(-.1,1.2),ylim=c(0,3.6), xlab="Beta",
main="Stratified QN, autosomes")
plotBetasByType(getBeta(StratQN)[,7], type_autosomes,
main="Stratified QN, Autosomes")
densityPlot(getBeta(StratQN_all),sampGroups=
Raw_preprocess@phenoData$Batch, pal = rainbow(3), xlim=c(-.
1,1.2),ylim=c(0,3.6), xlab="Beta", main="Stratified QN, all")
plotBetasByType(getBeta(StratQN_all)[,7], type_autosomes,
main="Stratified QN, all")

# FunNorm:  FunNorm, only have unprocessed all chr together,
separate?
par(mfrow=c(1,2))
densityPlot(FunNorm,sampGroups= FunNorm@phenoData$Batch, pal =
rainbow(3), xlim=c(-.1,1.2),ylim=c(0,3.6), xlab="Beta",
main="FunNorm")
plotBetasByType(FunNorm[,7], main="FunNorm by probe, all")

#SWAN: swan, swan_autosomes, swan_maleChr, swan_femaleChr
par(mfrow=c(1,4))
densityPlot(swan,sampGroups= swan@phenoData$Batch, pal = rainbow(3),
xlim=c(-.1,1.2),ylim=c(0,3.6), xlab="Beta", main="swan")
densityPlot(swan_autosomes,sampGroups= swan_autosomes@phenoData
$Batch, pal = rainbow(3), xlim=c(-.1,1.2),ylim=c(0,3.6),
xlab="Beta", main="swan_autosomes")
densityPlot(swan_maleChr,sampGroups= swan_maleChr@phenoData$Batch,
pal = rainbow(3), xlim=c(-.1,1.2),ylim=c(0,3.6), xlab="Beta",
main="swan_maleChr")
densityPlot(swan_femaleChr,sampGroups= swan_femaleChr@phenoData
$Batch, pal = rainbow(3), xlim=c(-.1,1.2),ylim=c(0,3.6),
xlab="Beta", main="Female_chr")
par(mfrow=c(1,4))
plotBetasByType(swan[,7], main="SWAN by probe, all")
plotBetasByType(swan_autosomes[,7], main="SWAN by probe, Autosomes")
plotBetasByType(swan_maleChr[,7], main="SWAN by probe, Male_chr")
plotBetasByType(swan_femaleChr[,7], main="SWAN by probe,
Female_chr")

# Dasen: Dasen, Dasen_autosomes, Dasen_male, Dasen_female
par(mfrow=c(1,4))
densityPlot(Dasen,sampGroups= Raw_preprocess@phenoData$Batch, pal =
rainbow(3), xlim=c(-.1,1.2),ylim=c(0,3.6), xlab="Beta",
main="Dasen")
densityPlot(Dasen_autosomes,sampGroups= Autosomes@phenoData$Batch,
pal = rainbow(3), xlim=c(-.1,1.2),ylim=c(0,3.6), xlab="Beta",
main="Dasen_autosomes")
densityPlot(Dasen_male,sampGroups= Male_chr@phenoData$Batch, pal =
rainbow(3), xlim=c(-.1,1.2),ylim=c(0,3.6), xlab="Beta",
```

```
main="Dasen_male")
densityPlot(Dasen_female,sampGroups= Female_chr@phenoData$Batch, pal
= rainbow(3), xlim=c(-.1,1.2),ylim=c(0,3.6), xlab="Beta",
main="Dasen_female")
# These last two graphs look really odd, again I think it's an issue
in calling the probe types

par(mfrow=c(1,4))
plotBetasByType(Dasen[,7], probeType=probeTypes_Dasen , main="Dasen
by probe, all")
plotBetasByType(Dasen_autosomes[,7],
probeType=probeTypes_autosomes2 , main="Dasen by probe, Autosomes")
# wont plot the last 2, maybe missing too many sites
plotBetasByType(Dasen_male[,7],
probeType=probeTypes_autosomes2_male, main="Dasen by probe,
Male_chr")
plotBetasByType(Dasen_female[,7],
probeType=probeTypes_autosomes2_female, main="Dasen by probe,
Female_chr")

#RUV: BetaRUV_autosomes, BetaRUV_autosomes_optimal, BetaRUV_male,
BetaRUV_female, BetaRUV_male_optimal, BetaRUV_female_optimal
par(mfrow=c(1,6))
densityPlot(BetaRUV_autosomes,sampGroups=Autosomes@phenoData$Batch,
pal = rainbow(3), xlim=c(-.1,1.2),ylim=c(0,20), xlab="Beta",
main="Density plot RUV data, Autosomes")
densityPlot(BetaRUV_autosomes_optimal,sampGroups=Autosomes@phenoData
$Batch, pal = rainbow(3), xlim=c(-.1,1.2),ylim=c(0,20), xlab="Beta",
main="Density plot RUV data, Autosomes_Optimal")
densityPlot(BetaRUV_male,sampGroups=Male_chr@phenoData$Batch, pal =
rainbow(3), xlim=c(-.1,1.2),ylim=c(0,20), xlab="Beta", main="Density
plot RUV data, Male_chr")
densityPlot(BetaRUV_female,sampGroups=Female_chr@phenoData$Batch,
pal = rainbow(3), xlim=c(-.1,1.2),ylim=c(0,20), xlab="Beta",
main="Density plot RUV data, Female_chr")
densityPlot(BetaRUV_male_optimal,sampGroups=Male_chr@phenoData
$Batch, pal = rainbow(3), xlim=c(-.1,1.2),ylim=c(0,20), xlab="Beta",
main="Density plot RUV data, Male_chr_optimal")
densityPlot(BetaRUV_female,sampGroups=Female_chr@phenoData$Batch,
pal = rainbow(3), xlim=c(-.1,1.2),ylim=c(0,20), xlab="Beta",
main="Density plot RUV data, Female_chr")

par(mfrow=c(1,6))
plotBetasByType(BetaRUV_autosomes[,1], type , main="Autosomes
BetaRUV by type", cex.legend=.8, legendPos="topright")
plotBetasByType(BetaRUV_autosomes_optimal[,1], type ,
main="Autosomes_optimal BetaRUV by type", cex.legend=.8,
legendPos="topright")
plotBetasByType(BetaRUV_male[,1], type_male , main="Male_chr BetaRUV
by type", cex.legend=.8, legendPos="topright")
plotBetasByType(BetaRUV_female[,1], type_female , main="Female_chr
BetaRUV by type", cex.legend=.8, legendPos="topright")
plotBetasByType(BetaRUV_male_optimal[,1], type_male ,
main="Male_chr_optimal BetaRUV by type", cex.legend=.8,
```

```
legendPos="topright")
plotBetasByType(BetaRUV_female_optimal[,1], type_female ,
main="Female_chr_optimal BetaRUV by type", cex.legend=.8,
legendPos="topright")

#QN: lumiQNbetas
par(mfrow=c(1,2))
densityPlot(lumiQNbetas,sampGroups=lumiQN_prepro@phenoData$Batch,
pal = rainbow(3), xlim=c(-.1,1.2),ylim=c(0,20), xlab="Beta",
main="Density plot lumiQN data, All chr")
plotBetasByType(lumiQNbetas[,1], probeTypes=type_methylumi,
main="lumiQNbetas by probe")

#BMIQ: BMIQ
par(mfrow=c(1,2))
densityPlot(BMIQ_betas,sampGroups=BMIQ@phenoData$Batch, pal =
rainbow(3), xlim=c(-.1,1),ylim=c(0,4), xlab="Beta", main="Density
plot BMIQ data, All chr")
plotBetasByType(BMIQ_betas[,1], probeTypes=type_BMIQ, main="BMIQ by
probe")

#Limma and RUV
par(mfrow=c(1,2))
densityPlot(BetaRUV_limma,sampGroups=Autosomes@phenoData$Batch, pal
= rainbow(3), xlim=c(-.1,1.2),ylim=c(0,20), xlab="Beta",
main="Density plot BetaRUV_limma, Autosomes")
densityPlot(BetaRUV_limma_optimal,sampGroups=Autosomes@phenoData
$Batch, pal = rainbow(3), xlim=c(-.1,1.2),ylim=c(0,3.5),
xlab="Beta", main="Density plot BetaRUV_limma_optimal, Autosomes")
par(mfrow=c(1,2))
plotBetasByType(BetaRUV_limma[,1], probeTypes=probeTypes_autosomes2,
main="BetaRUV_limma, Autosomes by probe")
plotBetasByType(BetaRUV_limma_optimal[,1],
probeTypes=probeTypes_autosomes2, main="BetaRUV_limma_optimal,
Autosomes by probe")

#Combat
par(mfrow=c(1,2))
densityPlot(ComBat_autosomes,sampGroups=Autosomes@phenoData$Batch,
pal = rainbow(3), xlim=c(-.1,1.2),ylim=c(0,20), xlab="Beta",
main="Density plot Combat, Autosomes")
plotBetasByType(ComBat_autosomes[,1],
probeTypes=probeTypes_autosomes2, main="Combat, Autosomes by probe")



------------------------------------------------------------------------
-------------------------

####################
# 2. MDS plots   ###
####################

# Raw: Raw_preprocess, Autosomes, Male_chr, Female_chr
```

```
par(mfrow=c(1,4))
mdsPlot(Raw_preprocess, numPositions=1000,
sampNames=Raw_preprocess@phenoData$Sample_Name,
sampGroups=Raw_preprocess@phenoData$Batch, xlim=c(-16,16),
ylim=c(-7,5),legendNCol=5, legendPos = "bottom", main="MDS-
Raw_preprocess")
mdsPlot(Autosomes, numPositions=1000, sampNames=Autosomes@phenoData
$Sample_Name, sampGroups=Autosomes@phenoData$Batch, xlim=c(-16,16),
ylim=c(-7,5),legendNCol=5, legendPos = "bottom", main="MDS- Raw
Autosomes")
mdsPlot(Male_chr, numPositions=1000, sampNames=Male_chr@phenoData
$Sample_Name, sampGroups=Male_chr@phenoData$Batch, xlim=c(-16,16),
ylim=c(-7,5),legendNCol=5, legendPos = "bottom", main="MDS-
Male_chr")
mdsPlot(Female_chr, numPositions=1000,
sampNames=Female_chr@phenoData$Sample_Name,
sampGroups=Female_chr@phenoData$Batch, xlim=c(-16,16),
ylim=c(-7,5),legendNCol=5, legendPos = "bottom", main="MDS-
Female_chr")

# StratQN
par(mfrow=c(1,2))
mdsPlot(getBeta(StratQN), numPositions=1000, sampNames=
Autosomes@phenoData$Sample_Name, sampGroups= Autosomes@phenoData
$Batch, xlim=c(-16,16), ylim=c(-7,5),legendNCol=5, legendPos =
"bottom", main="MDS- StratQN")
mdsPlot(getBeta(StratQN_all), numPositions=1000, sampNames=
Raw_preprocess@phenoData$Sample_Name, sampGroups=
Raw_preprocess@phenoData$Batch, xlim=c(-16,16),
ylim=c(-7,5),legendNCol=5, legendPos = "bottom", main="MDS-
StratQN_all")

# FunNorm: Raw_preprocess, need to figure out how to separate sex
chr
mdsPlot(FunNorm_betas, numPositions=1000,
sampNames=RGsetE2@phenoData$Sample_Name,
sampGroups=RGsetE2@phenoData$Batch, xlim=c(-16,16),
ylim=c(-7,5),legendNCol=5, legendPos = "bottom", main="MDS-
FunNorm")

# SWAN: swan, swan_autosomes, swan_maleChr, swan_femaleChr
par(mfrow=c(1,4))
mdsPlot(swan, numPositions=1000, sampNames=swan@phenoData
$Sample_Name, sampGroups=swan@phenoData$Batch, xlim=c(-16,16),
ylim=c(-7,5),legendNCol=5, legendPos = "bottom", main="MDS- SWAN")
mdsPlot(swan_autosomes, numPositions=1000, sampNames=
swan_autosomes@phenoData$Sample_Name, sampGroups=
swan_autosomes@phenoData$Batch, xlim=c(-16,16),
ylim=c(-7,5),legendNCol=5, legendPos = "bottom", main="MDS-
swan_autosomes")
mdsPlot(swan_maleChr, numPositions=1000, sampNames=
swan_maleChr@phenoData$Sample_Name, sampGroups=
swan_maleChr@phenoData$Batch, xlim=c(-16,16),
ylim=c(-7,5),legendNCol=5, legendPos = "bottom", main="MDS-
```

```
swan_maleChr")
mdsPlot(swan_femaleChr, numPositions=1000,
sampNames=swan_femaleChr@phenoData$Sample_Name,
sampGroups=swan_femaleChr@phenoData$Batch, xlim=c(-16,16),
ylim=c(-7,5),legendNCol=5, legendPos = "bottom", main="MDS-
swan_femaleChr")

# Dasen: Dasen,Dasen_autosomes, Dasen_male, Dasen_female
par(mfrow=c(1,4))
mdsPlot(Dasen, numPositions=1000, sampNames=
Raw_preprocess@phenoData$Sample_Name, sampGroups=
Raw_preprocess@phenoData$Batch, xlim=c(-16,16),
ylim=c(-7,5),legendNCol=5, legendPos = "bottom", main="MDS- Dasen")
mdsPlot(Dasen_autosomes, numPositions=1000, sampNames=
Autosomes@phenoData$Sample_Name, sampGroups= Autosomes@phenoData
$Batch, xlim=c(-16,16), ylim=c(-7,5),legendNCol=5, legendPos =
"bottom", main="MDS- Dasen_autosomes")
mdsPlot(Dasen_male, numPositions=1000, sampNames= Male_chr@phenoData
$Sample_Name, sampGroups= Male_chr@phenoData$Batch, xlim=c(-16,16),
ylim=c(-7,5),legendNCol=5, legendPos = "bottom", main="MDS-
Dasen_male")
mdsPlot(Dasen_female, numPositions=1000, sampNames=
Female_chr@phenoData$Sample_Name, sampGroups= Female_chr@phenoData
$Batch, xlim=c(-16,16), ylim=c(-7,5),legendNCol=5, legendPos =
"bottom", main="MDS- Dasen_female")
# male and female separate with dasen doesn't work

# RUV: BetaRUV_autosomes, BetaRUV_autosomes_optimal, BetaRUV_male,
BetaRUV_female, BetaRUV_male_optimal, BetaRUV_female_optimal
par(mfrow=c(1,6))
mdsPlot(BetaRUV_autosomes, numPositions=1000, sampNames=
Autosomes@phenoData$Sample_Name, sampGroups= Autosomes@phenoData
$Batch, xlim=c(-16,16), ylim=c(-7,5),legendNCol=5, legendPos =
"bottom", main="MDS- BetaRUV_autosomes")
mdsPlot(BetaRUV_autosomes_optimal, numPositions=1000, sampNames=
Autosomes@phenoData$Sample_Name, sampGroups= Autosomes@phenoData
$Batch, xlim=c(-16,16), ylim=c(-7,5),legendNCol=5, legendPos =
"bottom", main="MDS- BetaRUV_autosomes_optimal")
mdsPlot(BetaRUV_male, numPositions=1000, sampNames=
Male_chr@phenoData$Sample_Name, sampGroups= Male_chr@phenoData
$Batch, xlim=c(-16,16), ylim=c(-7,5),legendNCol=5, legendPos =
"bottom", main="MDS- BetaRUV_male")
mdsPlot(BetaRUV_female, numPositions=1000, sampNames=
Female_chr@phenoData$Sample_Name, sampGroups= Female_chr@phenoData
$Batch, xlim=c(-16,16), ylim=c(-7,5),legendNCol=5, legendPos =
"bottom", main="MDS- BetaRUV_female")
mdsPlot(BetaRUV_male_optimal, numPositions=1000, sampNames=
Male_chr@phenoData$Sample_Name, sampGroups= Male_chr@phenoData
$Batch, xlim=c(-16,16), ylim=c(-7,5),legendNCol=5, legendPos =
"bottom", main="MDS- BetaRUV_male_optimal")
mdsPlot(BetaRUV_female_optimal, numPositions=1000, sampNames=
Female_chr@phenoData$Sample_Name, sampGroups= Female_chr@phenoData
$Batch, xlim=c(-16,16), ylim=c(-7,5),legendNCol=5, legendPos =
"bottom", main="MDS- BetaRUV_female_optimal")
```

```
# QN - lumiQNbetas, need to figure out how to separate sex chr
mdsPlot(lumiQNbetas, numPositions=1000, sampNames=
methylumi_prepro@phenoData$Sample_Name, sampGroups=
methylumi_prepro@phenoData$Batch, xlim=c(-16,16),
ylim=c(-7,5),legendNCol=5, legendPos = "bottom", main="MDS-
lumiQNbetas")

# BMIQ - BMIQ, need to figure out how to separate sex chr
mdsPlot(BMIQ_betas, numPositions=1000, sampNames=
methylumi_prepro@phenoData$Sample_Name, sampGroups=
methylumi_prepro@phenoData$Batch, xlim=c(-16,16),
ylim=c(-7,5),legendNCol=5, legendPos = "bottom", main="MDS- BMIQ")


------------------------------------------------------------------
------------

#########################################################
## 2b. ANOVA on the PCA of the 1000 variable sites ##
#########################################################

Raw_SD<- as.matrix(apply(Raw_manual_M,1,sd))
Raw_order<- Raw_SD[rev(order(Raw_SD)),]
Raw_Top1000<-as.matrix(Raw_order[1:1000])
Raw_Top1000_cpgs <- rownames(Raw_Top1000)

StratQN_M_SD<- as.matrix(apply(StratQN_M,1,sd))
StratQN_M_SD_order<- StratQN_M_SD[rev(order(StratQN_M_SD)),]
StratQN_M_SD_Top1000<-as.matrix(StratQN_M_SD_order[1:1000])
StratQN_M_SD_Top1000_cpgs <- rownames(StratQN_M_SD_Top1000)

match(Raw_Top1000_cpgs, StratQN_M_SD_Top1000_cpgs)  # some do
length(which(StratQN_M_SD_Top1000_cpgs %in% Raw_Top1000_cpgs)) # 774
match

Combat_StratQN_SD<- as.matrix(apply(Combat_StratQN_corrected_M,
1,sd))
Combat_StratQN_SD_order<-
Combat_StratQN_SD[rev(order(Combat_StratQN_SD)),]
Combat_StratQN_SD_Top1000<-
as.matrix(Combat_StratQN_SD_order[1:1000])
Combat_StratQN_SD_Top1000_cpgs <-
rownames(Combat_StratQN_SD_Top1000)

length(which(StratQN_M_SD_Top1000_cpgs %in%
Combat_StratQN_SD_Top1000_cpgs))  # 730
length(which(Raw_Top1000_cpgs %in% Combat_StratQN_SD_Top1000_cpgs))
#563

prcomp_Raw_Top1000 <- prcomp(t(Raw_manual_M[Raw_Top1000_cpgs,]))
anova_pc1_Raw1000_Family<- aov(prcomp_Raw_Top1000$x[,1] ~
factor(champ_data_mset_corrected@phenoData$Family))
summary(anova_pc1_Raw1000_Family)  # p-val   0.00143 **
```

```
prcomp_StratQN_Top1000 <-
prcomp(t(StratQN_M[StratQN_M_SD_Top1000_cpgs,]))
anova_pc1_StratQN1000_Family <- aov(prcomp_StratQN_Top1000$x[,1] ~
factor(champ_data_mset_corrected@phenoData$Family))
summary(anova_pc1_StratQN1000_Family)  # p-val  0.000381

prcomp_Combat_StratQN_Top1000 <-
prcomp(t(Combat_StratQN_corrected_M[Combat_StratQN_SD_Top1000_cpgs,]
))
anova_pc1_Combat_StratQN1000_Family <-
aov(prcomp_Combat_StratQN_Top1000$x[,1] ~
factor(champ_data_mset_corrected@phenoData$Family))
summary(anova_pc1_Combat_StratQN1000_Family)  # p-val  0.000795

# Now use the same method to look at how much clustering is due to
batch
anova_pc1_Raw1000_Batch <- aov(prcomp_Raw_Top1000$x[,1] ~
factor(champ_data_mset_corrected@phenoData$Batch))
summary(anova_pc1_Raw1000_Batch)  # p-val <2e-16 ***  highly sig

anova_pc1_StratQN1000_Batch <- aov(prcomp_StratQN_Top1000$x[,1] ~
factor(champ_data_mset_corrected@phenoData$Batch))
summary(anova_pc1_StratQN1000_Batch)  # p-val 6.04e-10 ***  less sig

anova_pc1_Combat_StratQN1000_Batch <-
aov(prcomp_Combat_StratQN_Top1000$x[,1] ~
factor(champ_data_mset_corrected@phenoData$Batch))
summary(anova_pc1_Combat_StratQN1000_Batch)  # p-val  0.97  NOT
SIGNIFICANT, this is what expected


# Table of ANOVA of PC1
aov_pval_table <- matrix(c(summary(anova_pc1_Raw_manual_M)[[1]]
[["Pr(>F)"]][[1]],summary(anova_pc1_M_QN)[[1]][["Pr(>F)"]][[1]],
summary(anova_pc1_StratQN_M)[[1]][["Pr(>F)"]][[1]],
summary(anova_pc1_BMIQ_M)[[1]][["Pr(>F)"]][[1]],
summary(anova_pc1_SWAN_M)[[1]][["Pr(>F)"]][[1]],
summary(anova_pc2_FunNorm_M)[[1]][["Pr(>F)"]][[1]],
summary(anova_pc1_Dasen_M)[[1]][["Pr(>F)"]][[1]],
summary(anova_pc1_Noob_M)[[1]][["Pr(>F)"]][[1]],
summary(anova_pc1_RUV_M)[[1]][["Pr(>F)"]][[1]],
summary(anova_pc1_RUV_limma_M)[[1]][["Pr(>F)"]][[1]],
summary(anova_pc1_ComBat_M)[[1]][["Pr(>F)"]][[1]],
summary(anova_pc1_ComBat_StratQN_M)[[1]][["Pr(>F)"]][[1]]), nrow=12,
ncol=1)
colnames(aov_pval_table)= c("P-val")
rownames(aov_pval_table)= c("Raw", "Qantile Normalisation",
"Stratified Quantile Normalisation", "BMIQ", "SWAN", "Functional
Normalisation", "Dasen", "Noob", "RUV", "RUV with limma", "Raw with
ComBat", "Stratified Quantile Normalisation with ComBat")
write.csv(aov_pval_table,file= "ANOVA pvals.csv")
```

```
------------------------------------------------------------------------
-----------


#########################################################################
#############
## 3. Unsupervised hierarchial clustering--- takes an excessive
amount of time ##
#########################################################################
#############

library(pvclust)

Raw_d_names <- getM(champ_data_mset_corrected)
colnames(Raw_d_names) <- champ_data_beta$mset@phenoData$Sample_Name
hc_Raw_names <- hclust(dist(t(Raw_d_names), method="euclidean"),
method="single")
StratQN_d_names <- getM(champ_StratQN)
colnames(StratQN_d_names) <- champ_data_beta$mset@phenoData
$Sample_Name
hc_StratQN_names <- hclust(dist(t(StratQN_d_names),
method="euclidean"), method="single")
StratQNcombat_d_names <- logit2(Combat_StratQN)
colnames(StratQNcombat_d_names) <- champ_data_beta$mset@phenoData
$Sample_Name
hc_StratQNcombat_names <- hclust(dist(t(StratQNcombat_d_names),
method="euclidean"), method="single")

par(mfrow=c(1,3))
plot(hc_Raw_names, main="Cluster Dendrogram by Batch:
Raw")
plot(hc_StratQN_names, main="Cluster Dendrogram by Batch:
Stratified QN")
plot(hc_StratQNcombat_names, main="Cluster Dendrogram by Batch:
Stratified QN, ComBat corrected")

## label by batch
Raw_d_batch <- getM(champ_data_mset_corrected)
colnames(Raw_d_batch) <- champ_data_beta$mset@phenoData$Batch
hc_Raw_batch <- hclust(dist(t(Raw_d_batch), method="euclidean"),
method="single")
StratQN_d_batch <- getM(champ_StratQN)
colnames(StratQN_d_batch) <- champ_data_beta$mset@phenoData$Batch
hc_StratQN_batch <- hclust(dist(t(StratQN_d_batch),
method="euclidean"), method="single")
StratQNcombat_d_batch <- logit2(Combat_StratQN)
colnames(StratQNcombat_d_batch) <- champ_data_beta$mset@phenoData
$Batch
hc_StratQNcombat_batch <- hclust(dist(t(StratQNcombat_d_batch),
method="euclidean"), method="single")

par(mfrow=c(3,1))
plot(hc_Raw_batch, main="Cluster Dendrogram by Batch:
```

```
Raw")
plot(hc_StratQN_batch, main="Cluster Dendrogram by Batch:
Stratified QN")
plot(hc_StratQNcombat_batch, main="Cluster Dendrogram by Batch:
Stratified QN, ComBat corrected")
-----------------------------------------------------------------
-----------



############################
## 4.   Replicate samples  ##
############################

# Density plots and MDS to visualise effects of normalisation then
compare the mean absolute diff in M values between diff replicates

# There are 3 different samples that have technical replicates that
passed QC:  22-16, 22-17, 72-213
Replicates <- Raw_preprocess[,c(6,27,8,28,31,41,16,18)]
# which batches/plates do they belong to?
Replicates@phenoData$Batch  # [1] 1 2 1 2 3 3 1 1
Replicates@phenoData$Plate  # [1] 1 3 1 3 4 5 2 2
Replicates_Mvals <- getM(Replicates)
colnames(Replicates_Mvals)=Replicates@phenoData$Sample_Name
#[1] "pc22-16"        "PC22-16 (AH9)"  "pc22-17"
#[4] "PC22-17 (AH10)" "PC22-17a"       "PC22-17b"
#[7] "pc72-213a"      "pc72-213b"

#  M = logit(Beta) = log( Meth / Unmeth )  This formula has problems
if either Meth or Unmeth is zero. For this reason, we can use
betaThreshold to make sure Beta is neither 0 nor 1, before taken the
logit. What makes sense for the offset and betaThreshold depends
crucially on how the data was preprocessed. Do not expect the
default values to be particular good.
# type
How are the values calculated? For getBeta setting type="Illumina"
sets offset=100 as per Genome Studio. For getM setting type=""
computes M-values as the logarithm of Meth/Unmeth, otherwise it is
computed as the logit of getBeta(object).
Replicates_Mvals2 <- getM(Replicates, type="", offset=100,
betaThreshold=TRUE)
summary(Replicates_Mvals2[,2])
# changing these argumenst does nothing to change the number of
infinite values!!

beta_default  <- getBeta(Replicates)
beta_illumina <- getBeta(Replicates, type="Illumina")
beta_threshold_100 <- getBeta(Replicates, offset=100)
beta_threshold_50 <- getBeta(Replicates, offset=50)

beta_illumina[1:10] # sets offset to 100 as genome studio does
beta_default[1:10]  # these values are slightly higher than illumina
and include 1.0000, ie no there is no offset and the M-values would
```

```
have issues
beta_threshold_100[1:10] # same as illumina
beta_threshold_50[1:10] # slightly higher values

Mvals_default <-getM(Replicates)  #logit of getBeta(object)
Mvals_default[1:10]  # this has inf for the 3rd value which was 1
before
Mvals_methUnmeth <- getM(Replicates, type="")  # logarithm of Meth/
Unmeth
Mvals_methUnmeth[1:10] # same as default, inf at 3
Mvals_threshold <- getM(Replicates, type="", betaThreshold=TRUE) #
same with or without type
Mvals_threshold[1:10]  # same as default, inf at 3 -- this shouldn't
be the case since threshold is specified... need to do it manually

# manually

Mvals_manual <- logit2(beta_default)
Mvals_manual[1:10]  #this is identical to Mvals_default, inf at 3
Mvals_manual_illumina <- logit2(beta_illumina)  ## use this
Mvals_manual_illumina[1:10]  # slightly lower values, the 3rd value
is not infinite
Mvals_manual_threshold100 <- logit2(beta_threshold_100)
Mvals_manual_threshold100[1:10] # identical to illumina one
Mvals_manual_threshold50 <- logit2(beta_threshold_50)
Mvals_manual_threshold50[1:10]  #slightly higher than threshold of
100

# Don't use the default beta values because they contain 1s, use:
Mvals_manual_illumina

logMethUnmeth <- logit2(getMeth(Replicates) / getUnmeth(Replicates))
logMethUnmeth[1:10]

beta_man <- getMeth(Replicates) / (getMeth(Replicates) +
getUnmeth(Replicates) + 100)  # maybe I need to keep adjusting the
offset so it makes the 0s big enough that their log is not -inf?? so
a sml threshold?   no :(   no it won't make a difference what the
threshold is if meth is 0 because the numerator will be 0... need to
add the offset to the top!  Beta = Meth / (Meth + Unmeth + offset)
beta_man[1:10]
beta_man_M <- logit2(beta_man)
length(which(beta_man_M[,1]==-Inf))

beta_man2 <- (getMeth(Replicates) +100) / (getMeth(Replicates) +
getUnmeth(Replicates) +100)
beta_man2[1:10]
beta_man_M2 <- logit2(beta_man2)
length(which(beta_man_M2[,1]==-Inf))  # yesssss this gets rid of the
negative infinity values but because adding the same number top and
bottom cancels out it creates + infinite values as beta can now be
equal to 1. If I add an offset of 50 to the top and 100 to the
bottom will this affect the value in a biased manner?
```

```
beta_man3 <- (getMeth(Replicates) +99.9) / (getMeth(Replicates) +
getUnmeth(Replicates) +100)
beta_man3[1:10]
beta_man_M3 <- logit2(beta_man3)
length(which(beta_man_M3[,1]==-Inf))  # yes both are length 0 :):):)
but has it changed values in a biased way.. if I make the difference
between them very small, say .1

# Use
Replicates_Mvals <- beta_man_M3

# look out for infinite values:
summary(Replicates_Mvals[,1])
#        Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
#       -7.6320 -3.0330  0.8257  0.3039  3.0100 17.6600
summary(Replicates_Mvals[,2])
#       Min. 1st   Qu.  Median    Mean 3rd Qu.    Max.
#       -7.2020 -2.9490  0.9983  0.3781  3.1330 16.9400
summary(Replicates_Mvals[,3])
#   Min.  1st Qu.  Median    Mean 3rd Qu.    Max.
#   -7.7050 -3.0970  0.9217  0.3675  3.1950 17.9200
summary(Replicates_Mvals[,4])
#  Min.  1st Qu.  Median    Mean 3rd Qu.    Max.
#  -7.2920 -3.1900  1.0640  0.3447  3.2780 17.2000
summary(Replicates_Mvals[,5])
#   Min.  1st Qu.  Median    Mean 3rd Qu.    Max.
#  -8.87800 -3.13200  0.63340  0.08353  2.63400 18.76000
summary(Replicates_Mvals[,6])
#    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
#   -7.9550 -2.9350  0.7339  0.2195  2.8100 18.1800
summary(Replicates_Mvals[,7])
#   Min.  1st Qu.  Median    Mean 3rd Qu.    Max.
#  -7.3840 -2.8220  0.9293  0.3257  3.0050 17.4500
summary(Replicates_Mvals[,8])
#   Min.  1st Qu.  Median    Mean 3rd Qu.    Max.
#   -7.4520 -2.7940  0.9521  0.3328  3.0040 17.4400

# how is the data diustributed?  # skewed to the left
tail(Replicates_Mvals[,1])
hist(Replicates_Mvals[,1])
tail(Replicates_Mvals[,2])
hist(Replicates_Mvals[,2])
tail(Replicates_Mvals[,3])
hist(Replicates_Mvals[,3])
tail(Replicates_Mvals[,4])
hist(Replicates_Mvals[,4])
tail(Replicates_Mvals[,5])
hist(Replicates_Mvals[,5])
tail(Replicates_Mvals[,6])
hist(Replicates_Mvals[,6])
tail(Replicates_Mvals[,7])
hist(Replicates_Mvals[,7])
tail(Replicates_Mvals[,8])
hist(Replicates_Mvals[,8])
```

```
######
# 1: "pc22-16"            batch 1, chip 1
# 2: "PC22-16 (AH9)"      batch 2, chip 3
# 3: "pc22-17"                    batch 1, chip 1
# 4: "PC-17 (AH10)"       batch 2, chip 3
# 5: "PC22-17a"           batch 3, chip 4
# 6: "PC22-17b"           batch 3, chip 5
# 7: "pc72-213a"          batch 1, chip 2
# 8: "pc72-213b"          batch 1, chip 2


# Compare:

# 1. 22.16 (1 & 2)        diff plate, diff batch (one OS, one here)
"pc22-16"        "PC22-16 (AH9)"
# 2. 72.213 (7 & 8)       same plate, same batch, different
sample(array) : Expect to be smallest diff
22.17 (3,4,5,6)           all different plates, 3/4 different
batches
# 3. 22.17 (3,4)          diff plate, diff batch
# 4. 22.17 (3,5)                  diff plate, diff batch
# 5. 22.17 (3,6)                  diff plate, diff batch
# 6. 22.17 (4,5)                  diff plate, diff batch
# 7. 22.17 (4,6)          diff plate, diff batch
# 8. 22.17 (5,6)                  diff plate, same batch  : expect
middle diff

# 9. (1 & 3) compare diff samples same batch, same plate
#10. (3,7) diff sample, same batch , diff plate
#11. (6,7) diff sample, diff batch, diff plate


#1.
#MeanDiff_22.16=mean(abs(Replicates_Mvals[,1] - Replicates_Mvals[,
2])) #  0.6772961
MeanDiffb_22.16=abs(mean(Replicates_Mvals[,1]) -
mean(Replicates_Mvals[,2])) # 0.07418344
# which way?

#2
MeanDiffb_72.213=abs(mean(Replicates_Mvals[,7]) -
mean(Replicates_Mvals[,8]))       # 0.007016831
#3
MeanDiffb_22.17=abs(mean(Replicates_Mvals[,3]) -
mean(Replicates_Mvals[,4]))       # 0.02274291
#4
MeanDiffb_22.17=abs(mean(Replicates_Mvals[,3]) -
mean(Replicates_Mvals[,5]))       # 0.2839539
#5
MeanDiffb_22.17=abs(mean(Replicates_Mvals[,3]) -
mean(Replicates_Mvals[,6]))       # 0.1479692
#6
MeanDiffb_22.17=abs(mean(Replicates_Mvals[,4]) -
```

```
mean(Replicates_Mvals[,5]))       # 0.261211
#7
MeanDiffb_22.17=abs(mean(Replicates_Mvals[,4]) -
mean(Replicates_Mvals[,6]))       # 0.1252263
#8
MeanDiffb_22.17=abs(mean(Replicates_Mvals[,5]) -
mean(Replicates_Mvals[,6]))       # 0.1359847
-------
#9
MeanDiffb_batch1 <- abs(mean(Replicates_Mvals[,1])-
mean(Replicates_Mvals[,3]))       # 0.06358556
#10
MeanDiffb_plate <- abs(mean(Replicates_Mvals[,3])-
mean(Replicates_Mvals[,7]))       # 0.04174628
#11
MeanDiffb_plate <- abs(mean(Replicates_Mvals[,6])-
mean(Replicates_Mvals[,7]))       # 0.1062229

# these should be much bigger differences than the diff between the
samples but aren't.  THis is preprocessed data so batch effects
haven't been removed yet, thus makes sense

#1  rep 22.16:  diff plate, diff batch
table_replicate_medians_rep16 <- matrix(c(medianDiff_22.16_raw,
medianDiff_22.16_QN, medianDiff_22.16_champ_StratQN,
medianDiff_22.16_BMIQ, medianDiff_22.16_SWAN,
medianDiff_22.16_FunNorm, medianDiff_22.16_Dasen,
medianDiff_22.16_Noob, medianDiff_22.16_BetaRUV,
medianDiff_22.16_BetaRUV_limma, medianDiff_22.16_ComBat,
medianDiff_22.16_Combat_StratQN), ncol=1)
rownames(table_replicate_medians_rep16) <- c("Raw", "Qantile
Normalisation", "Stratified Quantile Normalisation", "BMIQ", "SWAN",
"Functional Normalisation", "Dasen", "Noob", "RUV", "RUV with
limma", "Raw with ComBat", "Stratified Quantile Normalisation with
ComBat")
colnames(table_replicate_medians_rep16) <- "Median absolute
difference between replicates"
write.csv(table_replicate_medians_rep16, file="Replicate.1 Median
Diffs.csv")

#2  72.213:  same batch, same plate
table_replicate_medians_rep72.213 <- matrix(c(medianDiff_72.213_raw,
medianDiff_72.213_QN, medianDiff_72.213_champ_StratQN,
medianDiff_72.213_BMIQ, medianDiff_72.213_SWAN,
medianDiff_72.213_FunNorm, medianDiff_72.213_Dasen,
medianDiff_72.213_Noob, medianDiff_72.213_BetaRUV,
medianDiff_72.213_BetaRUV_limma, medianDiff_72.213_ComBat,
medianDiff_72.213_Combat_StratQN), ncol=1)
rownames(table_replicate_medians_rep72.213) <- c("Raw", "Qantile
Normalisation", "Stratified Quantile Normalisation", "BMIQ", "SWAN",
"Functional Normalisation", "Dasen", "Noob", "RUV", "RUV with
limma", "Raw with ComBat", "Stratified Quantile Normalisation with
ComBat")
colnames(table_replicate_medians_rep72.213) <- "Median absolute
```

```
difference between replicates"
write.csv(table_replicate_medians_rep72.213, file="Replicate.2
Median Diffs.csv")

#3 22.17a:  diff plate, diff batch
table_replicate_medians_rep22.17a <- matrix(c(medianDiff_22.17a_raw,
medianDiff_22.17a_QN, medianDiff_22.17a_champ_StratQN,
medianDiff_22.17a_BMIQ, medianDiff_22.17a_SWAN,
medianDiff_22.17a_FunNorm, medianDiff_22.17a_Dasen,
medianDiff_22.17a_Noob, medianDiff_22.17a_BetaRUV,
medianDiff_22.17a_BetaRUV_limma, medianDiff_22.17a_ComBat,
medianDiff_22.17a_Combat_StratQN), ncol=1)
rownames(table_replicate_medians_rep22.17a) <- c("Raw", "Qantile
Normalisation", "Stratified Quantile Normalisation", "BMIQ", "SWAN",
"Functional Normalisation", "Dasen", "Noob", "RUV", "RUV with
limma", "Raw with ComBat", "Stratified Quantile Normalisation with
ComBat")
colnames(table_replicate_medians_rep22.17a) <- "Median absolute
difference between replicates"
write.csv(table_replicate_medians_rep22.17a, file="Replicate.3
Median Diffs.csv")

#4  22.17b
table_replicate_medians_rep22.17b <- matrix(c(medianDiff_22.17b_raw,
medianDiff_22.17b_QN, medianDiff_22.17b_champ_StratQN,
medianDiff_22.17b_BMIQ, medianDiff_22.17b_SWAN,
medianDiff_22.17b_FunNorm, medianDiff_22.17b_Dasen,
medianDiff_22.17b_Noob, medianDiff_22.17b_BetaRUV,
medianDiff_22.17b_BetaRUV_limma, medianDiff_22.17b_ComBat,
medianDiff_22.17b_Combat_StratQN), ncol=1)
rownames(table_replicate_medians_rep22.17b) <- c("Raw", "Qantile
Normalisation", "Stratified Quantile Normalisation", "BMIQ", "SWAN",
"Functional Normalisation", "Dasen", "Noob", "RUV", "RUV with
limma", "Raw with ComBat", "Stratified Quantile Normalisation with
ComBat")
colnames(table_replicate_medians_rep22.17b) <- "Median absolute
difference between replicates"
write.csv(table_replicate_medians_rep22.17b, file="Replicate.4
Median Diffs.csv")

#5 22.17c
table_replicate_medians_rep22.17c <- matrix(c(medianDiff_22.17c_raw,
medianDiff_22.17c_QN, medianDiff_22.17c_champ_StratQN,
medianDiff_22.17c_BMIQ, medianDiff_22.17c_SWAN,
medianDiff_22.17c_FunNorm, medianDiff_22.17c_Dasen,
medianDiff_22.17c_Noob, medianDiff_22.17c_BetaRUV,
medianDiff_22.17c_BetaRUV_limma, medianDiff_22.17c_ComBat,
medianDiff_22.17c_Combat_StratQN), ncol=1)
rownames(table_replicate_medians_rep22.17c) <- c("Raw", "Qantile
Normalisation", "Stratified Quantile Normalisation", "BMIQ", "SWAN",
"Functional Normalisation", "Dasen", "Noob", "RUV", "RUV with
limma", "Raw with ComBat", "Stratified Quantile Normalisation with
ComBat")
colnames(table_replicate_medians_rep22.17c) <- "Median absolute
```

```
difference between replicates"
write.csv(table_replicate_medians_rep22.17c, file="Replicate.5
Median Diffs.csv")

#6 22.17d
table_replicate_medians_rep22.17d <- matrix(c(medianDiff_22.17d_raw,
medianDiff_22.17d_QN, medianDiff_22.17d_champ_StratQN,
medianDiff_22.17d_BMIQ, medianDiff_22.17d_SWAN,
medianDiff_22.17d_FunNorm, medianDiff_22.17d_Dasen,
medianDiff_22.17d_Noob, medianDiff_22.17d_BetaRUV,
medianDiff_22.17d_BetaRUV_limma, medianDiff_22.17d_ComBat,
medianDiff_22.17d_Combat_StratQN), ncol=1)
rownames(table_replicate_medians_rep22.17d) <- c("Raw", "Qantile
Normalisation", "Stratified Quantile Normalisation", "BMIQ", "SWAN",
"Functional Normalisation", "Dasen", "Noob", "RUV", "RUV with
limma", "Raw with ComBat", "Stratified Quantile Normalisation with
ComBat")
colnames(table_replicate_medians_rep22.17d) <- "Median absolute
difference between replicates"
write.csv(table_replicate_medians_rep22.17d, file="Replicate.6
Median Diffs.csv")

#7 22.17e
table_replicate_medians_rep22.17e <- matrix(c(medianDiff_22.17e_raw,
medianDiff_22.17e_QN, medianDiff_22.17e_champ_StratQN,
medianDiff_22.17e_BMIQ, medianDiff_22.17e_SWAN,
medianDiff_22.17e_FunNorm, medianDiff_22.17e_Dasen,
medianDiff_22.17e_Noob, medianDiff_22.17e_BetaRUV,
medianDiff_22.17e_BetaRUV_limma, medianDiff_22.17e_ComBat,
medianDiff_22.17e_Combat_StratQN), ncol=1)
rownames(table_replicate_medians_rep22.17e) <- c("Raw", "Qantile
Normalisation", "Stratified Quantile Normalisation", "BMIQ", "SWAN",
"Functional Normalisation", "Dasen", "Noob", "RUV", "RUV with
limma", "Raw with ComBat", "Stratified Quantile Normalisation with
ComBat")
colnames(table_replicate_medians_rep22.17e) <- "Median absolute
difference between replicates"
write.csv(table_replicate_medians_rep22.17e, file="Replicate.7
Median Diffs.csv")

#8 22.17f
table_replicate_medians_rep22.17f <- matrix(c(medianDiff_22.17f_raw,
medianDiff_22.17f_QN, medianDiff_22.17f_champ_StratQN,
medianDiff_22.17f_BMIQ, medianDiff_22.17f_SWAN,
medianDiff_22.17f_FunNorm, medianDiff_22.17f_Dasen,
medianDiff_22.17f_Noob, medianDiff_22.17f_BetaRUV,
medianDiff_22.17f_BetaRUV_limma, medianDiff_22.17f_ComBat,
medianDiff_22.17f_Combat_StratQN), ncol=1)
rownames(table_replicate_medians_rep22.17f) <- c("Raw", "Qantile
Normalisation", "Stratified Quantile Normalisation", "BMIQ", "SWAN",
"Functional Normalisation", "Dasen", "Noob", "RUV", "RUV with
limma", "Raw with ComBat", "Stratified Quantile Normalisation with
ComBat")
colnames(table_replicate_medians_rep22.17f) <- "Median absolute
```

```
difference between replicates"
write.csv(table_replicate_medians_rep22.17f, file="Replicate.8
Median Diffs.csv")


# 9. DiffSample1: diff samples, same batch, same plate  diffsamp1
table_replicate_medians_diffsamp1 <-
matrix(c(medianDiff_diffsamp1_raw, medianDiff_diffsamp1_QN,
medianDiff_diffsamp1_champ_StratQN, medianDiff_diffsamp1_BMIQ,
medianDiff_diffsamp1_SWAN, medianDiff_diffsamp1_FunNorm,
medianDiff_diffsamp1_Dasen, medianDiff_diffsamp1_Noob,
medianDiff_diffsamp1_BetaRUV, medianDiff_diffsamp1_BetaRUV_limma,
medianDiff_diffsamp1_ComBat, medianDiff_diffsamp1_Combat_StratQN),
ncol=1)
rownames(table_replicate_medians_diffsamp1) <- c("Raw", "Qantile
Normalisation", "Stratified Quantile Normalisation", "BMIQ", "SWAN",
"Functional Normalisation", "Dasen", "Noob", "RUV", "RUV with
limma", "Raw with ComBat", "Stratified Quantile Normalisation with
ComBat")
colnames(table_replicate_medians_diffsamp1) <- "Median absolute
difference between replicates"
write.csv(table_replicate_medians_diffsamp1, file="DiffSample.1
Median Diffs.csv")
#10. DiffSample2: diff sample, same batch , diff plate  diffsamp2
table_replicate_medians_diffsamp2 <-
matrix(c(medianDiff_diffsamp2_raw, medianDiff_diffsamp2_QN,
medianDiff_diffsamp2_champ_StratQN, medianDiff_diffsamp2_BMIQ,
medianDiff_diffsamp2_SWAN, medianDiff_diffsamp2_FunNorm,
medianDiff_diffsamp2_Dasen, medianDiff_diffsamp2_Noob,
medianDiff_diffsamp2_BetaRUV, medianDiff_diffsamp2_BetaRUV_limma,
medianDiff_diffsamp2_ComBat, medianDiff_diffsamp2_Combat_StratQN),
ncol=1)
rownames(table_replicate_medians_diffsamp2) <- c("Raw", "Qantile
Normalisation", "Stratified Quantile Normalisation", "BMIQ", "SWAN",
"Functional Normalisation", "Dasen", "Noob", "RUV", "RUV with
limma", "Raw with ComBat", "Stratified Quantile Normalisation with
ComBat")
colnames(table_replicate_medians_diffsamp2) <- "Median absolute
difference between replicates"
write.csv(table_replicate_medians_diffsamp2, file="DiffSample.2
Median Diffs.csv")
#11. DiffSample3: diff sample, diff batch, diff plate   diffsamp3
table_replicate_medians_diffsamp3 <-
matrix(c(medianDiff_diffsamp3_raw, medianDiff_diffsamp3_QN,
medianDiff_diffsamp3_champ_StratQN, medianDiff_diffsamp3_BMIQ,
medianDiff_diffsamp3_SWAN, medianDiff_diffsamp3_FunNorm,
medianDiff_diffsamp3_Dasen, medianDiff_diffsamp3_Noob,
medianDiff_diffsamp3_BetaRUV, medianDiff_diffsamp3_BetaRUV_limma,
medianDiff_diffsamp3_ComBat, medianDiff_diffsamp3_Combat_StratQN),
ncol=1)
rownames(table_replicate_medians_diffsamp3) <- c("Raw", "Qantile
Normalisation", "Stratified Quantile Normalisation", "BMIQ", "SWAN",
"Functional Normalisation", "Dasen", "Noob", "RUV", "RUV with
limma", "Raw with ComBat", "Stratified Quantile Normalisation with
```

```
ComBat")
colnames(table_replicate_medians_diffsamp3) <- "Median absolute
difference between replicates"
write.csv(table_replicate_medians_diffsamp3, file="DiffSample.3
Median Diffs.csv")



########################
### 5. Dasen metrics ###
########################
# iDMRs, XCI, 65snps - standard error type measurement for each
(DMRSE, 1-AUC, GCOSE)
# only iDMRs work on this data


library(wateRmelon)
############
# a) iDMRs:
# Imprinting differentially methylated regions (iDMRs) are expected
to be approximately half methylated, as is observed at the 227
probes in known iDMRs. These functions calculate measures of
dispersion for the beta values at these CpG sites,
# Returns a standard error of the mean of betas for all samples and
iDMR probes (dmrse) or the standard error of the mean for just the
between sample component (dmrse_row) or between probe(dmrse_col)
component.
# lower values are better

dmrse_row(Raw_preprocess)  / dmrse_row(Autosomes)
# 227 iDMR data rows found
# [1] 0.004897683
dmrse_row(Male_chr) / dmrse_row(Female_chr) # don't need all probes
as autosomes works but can't just do on sex probes
# 0 iDMR data rows found
dmrse_row(StratQN_betas)                              # 0.003342732
dmrse_row(FunNorm_betas)                              # 0.005589773
dmrse_row(swan)                                       # 0.004925195
dmrse_row(swan_autosomes)                             # 0.004920393
dmrse_row(Dasen)                                              #
0.004927997
dmrse_row(Dasen_autosomes)                            # 0.004956462
dmrse_row(BetaRUV_autosomes)                 # 0.00422788
dmrse_row(BetaRUV_autosomes_optimal) # 0.00489734  worse than
default setting
dmrse_row(lumiQNbetas)                                # 0.005220801
dmrse_row(BMIQ)                                               #
0.006240381
dmrse_row(ComBat_autosomes)                  # 0.002518785

# Stratified QN looks to be the best normalisation method. Combat
batch correction on preprocessed data is the overall best.
# Try combat on other norm methods if possible
```

```
#############
# b) XCI - seabi
# Can only use on the objects normalised with sex CHR or just men

# Calculates an area under ROC curve - based metric for Illumina
450K data using a t-test for male-female difference as the predictor
for X-chromosome location of probes. The metric is 1-area so that
small values indicate good performance,to match our other, standard
error based metrics gcose and dmrse. Note that this requires both
male and female samples of known sex and can be slow to compute due
to running a t-test on every probe.

seabird_Raw<- seabi(getBeta(Raw_preprocess), sex=
Raw_preprocess@phenoData$Sex, X=as.logical(Annotated@listData
$chr=="chrX")) # takes awhile but computes in the end
#[1] 0.09567028
seabird_StratQN<- seabi(getBeta(StratQN_all), sex=
Raw_preprocess@phenoData$Sex, X=as.logical(Annotated@listData
$chr=="chrX"))
#
seabird_StratQN<- seabi(getBeta(swan), sex= swan@phenoData$Sex,
X=as.logical(Annotated@listData$chr=="chrX"))
#

#############
# c) 65 SNPs:
#There are 65 well-behaved SNP genotyping probes included on the
array. These each produce a distribution of betas with tight peaks
for the three possible genotypes, which will be broadened by
technical variation between samples. The spread of the peaks is thus
usable as a performance metric.
# genki: A very simple genotype calling by one-dimensional K-means
clustering is performed on each SNP, and for those SNPs where there
are three genotypes, the squared deviations are summed for each
genotype (similar to a standard deviation for each of allele A
homozygote, heterozygote and allele B homozygote). By default these
are further divided by the square root of the number of samples to
get a standard error-like statistic.
# should get: a vector of 3 values for the dispersion of the three
genotype peaks (AA, AB, BB : low, medium and high beta values)
#gcose - calculate between-sample SNP standard error

genki_Autosomes <- genki(getBeta(Autosomes),
g=getsnp(rownames(Autosomes)), se=TRUE)
#0SNP data rows found, Error in x[1:3, ] : incorrect number of
dimensions

y <- preprocessRaw(RGsetE2)
rownames(y)
x <- grep("rs", rownames(y), ignore.case=TRUE,)

UNABLE TO FIND SNP PROBES
```

```
------------------------------------------------------------------------
-----------

#########################
### 6. positive controls ###
#########################

# CpGs known to affect methylation, use minfi function 'plotCpg'

# CpG-SNP sites with most significant phenotype-wide correlated cis
associations - in the brain, fromZhang etc al 2010

library(minfi)
cpgs=c("cg24920358","cg22333868","cg13926569", "cg17749961",
"cg10106388", "cg06873352", "cg13507326", "cg01561916",
"cg14141399", "cg18294158")

# par(mfrow=c(2,5))
# plotCpg(champ_raw_beta,cpg=cpgs, pheno=champ_data_beta
$mset@phenoData$Batch, ylim=c(0,1), mainPrefix="Raw")
# plotCpg(getBeta(champ_StratQN),cpg=cpgs, pheno=champ_data_beta
$mset@phenoData$Batch, ylim=c(0,1), mainPrefix="StratQN")
# plotCpg(Combat_StratQN_corrected,cpg=cpgs, pheno=champ_data_beta
$mset@phenoData$Batch, ylim=c(0,1), mainPrefix="StratQN with
Combat")

# plotCpg(champ_raw_beta,cpg=cpgs, pheno=champ_data_beta
$mset@phenoData$Family, ylim=c(0,1), mainPrefix="Raw")
# plotCpg(getBeta(champ_StratQN),cpg=cpgs, pheno=champ_data_beta
$mset@phenoData$Family, ylim=c(0,1), mainPrefix="StratQN")
# plotCpg(Combat_StratQN_corrected,cpg=cpgs, pheno=champ_data_beta
$mset@phenoData$Family, ylim=c(0,1), mainPrefix="StratQN with
Combat")

Raw_zhang <- Raw_manual_M[cpgs,]
StratQN_zhang <- StratQN_M[cpgs,]
Combat_StratQN_zhang <- Combat_StratQN_corrected_M[cpgs,]

colnames(Raw_zhang) <- champ_data_beta$mset@phenoData$Sample_Name
Raw_zhang <- t(Raw_zhang)
row.names(Raw_zhang)  <- gsub("..AH[1-9]+.","",row.names(Raw_zhang))
row.names(Raw_zhang)  <-
gsub("PC22-17)","PC22-17",row.names(Raw_zhang))
row.names(Raw_zhang)  <- gsub("a","",row.names(Raw_zhang))
row.names(Raw_zhang)  <- gsub("b","",row.names(Raw_zhang))
row.names(Raw_zhang)  <- gsub("pc","PC",row.names(Raw_zhang))
row.names(Raw_zhang)  <- gsub("-",".",row.names(Raw_zhang))
mode(Raw_zhang) <- "numeric"
par(mar=c(6,4,1,1))
boxplot(Raw_zhang,las=2)

colnames(StratQN_zhang) <- champ_data_beta$mset@phenoData
```

```
$Sample_Name
StratQN_zhang <- t(StratQN_zhang)
row.names(StratQN_zhang)  <-
gsub("..AH[1-9]+.","",row.names(StratQN_zhang))
row.names(StratQN_zhang)  <-
gsub("PC22-17)","PC22-17",row.names(StratQN_zhang))
row.names(StratQN_zhang)  <- gsub("a","",row.names(StratQN_zhang))
row.names(StratQN_zhang)  <- gsub("b","",row.names(StratQN_zhang))
row.names(StratQN_zhang)  <-
gsub("pc","PC",row.names(StratQN_zhang))
row.names(StratQN_zhang)  <- gsub("-",".",row.names(StratQN_zhang))

mode(StratQN_zhang) <- "numeric"
par(mar=c(6,4,1,1))
boxplot(StratQN_zhang,las=2)

colnames(Combat_StratQN_zhang) <- champ_data_beta$mset@phenoData
$Sample_Name
Combat_StratQN_zhang <- t(Combat_StratQN_zhang)
row.names(Combat_StratQN_zhang) <- row.names(StratQN_zhang)

mode(Combat_StratQN_zhang) <- "numeric"
par(mar=c(6,4,1,1))
boxplot(Combat_StratQN_zhang,las=2)

### Replace this with new meth data, ie. will need to repeat loop 3x
Raw_zhang
StratQN_zhang
Combat_StratQN_zhang

chr <- c(1,12,10,2,1,17,14,2,19,7)
pos <-
c(39991652,78121579,89408710,30530779,159099485,59216916,77233176,42
876472,56919728,103657263)
cpgnames <- colnames(Raw_zhang)

for (cpg in 1:length(chr)) {
        system(paste(c("/Applications/plink-1.07-mac-intel/plink --
noweb --file ~/Documents/GENEPI/prostate/illumina/PCillumina/data --
chr ",chr[cpg]," --from-bp ",pos[cpg]-10^6," --to-bp ",pos[cpg]
+10^6," --recode --out ~/Users/ecazaly/Desktop/R Dec14/NormPaper/
cpg",cpg),collapse=""))
}
#for (cpg in 1:length(chr)) {
         cpg=4
        system(paste(c("/Applications/plink-1.07-mac-intel/plink --
noweb --file /Users/ecazaly/Desktop/R\\ Dec14/NormPaper/
PLINK_PCnoerrors_MAF05/PCnoerrors_MAF05 --chr ",chr[cpg]," --from-bp
",pos[cpg]-10^6," --to-bp ",pos[cpg]+10^6," --recode --out /Users/
ecazaly/Desktop/R\\ Dec14/NormPaper/cpg",cpg),collapse=""))

for (cpg in 1:length(chr)) {

        PPIEgeno <- read.table("PPIE.ped",header=F)
```

```
        PPIEgeno <-
read.table(paste(c("cpg",cpg,".ped"),collapse=""),header=F,as.is=T)
        row.names(PPIEgeno) <-
gsub("Tas-","",as.character(PPIEgeno[,1]))
        row.names(PPIEgeno) <- gsub("_",".",row.names(PPIEgeno))
        row.names(PPIEgeno) <- gsub("Tas","",row.names(PPIEgeno))

        map <- read.table("PPIE.map",as.is=T)

        map <-
read.table(paste(c("cpg",cpg,".map"),collapse=""),as.is=T)
        nsnps <- nrow(map)
        log10pv <- vector(length=nsnps)

        for (i in 1:nsnps) {
                geno <- PPIEgeno[,(i*2+5):(i*2+6)]
                geno <- geno[rowSums(geno=="0")==0,]
                geno <-
geno[match(row.names(Raw_zhang),row.names(geno)),]
                als <- levels(factor(unlist(geno)))
                if (length(als)<=1 |
sum(geno[1]==geno[2],na.rm=T)==0) {
                        next
                }
                gc <- rowSums(geno==als[1])
                fit = lm(Raw_zhang[,cpg] ~ gc)   #need to account
for kinship
                pv <- summary(fit)$coef[2,4]
                log10pv[i] <- -1*log10(pv)
        }
?lm  # figure out where to put kinship as a covariate in model --
need to use the GenABEL model to account for kinship
# first need to create kinship object
png(paste(cpgnames[cpg],".png",sep=""),pointsize=12,units="mm",width
=137.6,height=137.6*2/3,res=800)
        par(family="serif")
        par(mar=c(4,4,3,0.5))
        plot(map[,
4]*10^-6,log10pv,type="b",main=cpgnames[cpg],xlab="Position
(Mb)",ylab="-1*log10(p-value)",xlim=c(map[1,4]*10^-6,map[nsnps,
4]*10^-6+0.4),ylim=c(-0.5,4))
        abline(h=-1*log10(0.05),lty=2)
        text(map[nsnps,4]*10^-6+0.15,-1*log10(0.05)+0.13,"p-value =
0.05",font=3)
        abline(h=-1*log10(0.05/nsnps),lty=2)
        text(map[nsnps,4]*10^-6+0.15,-1*log10(0.05/nsnps)
+0.13,"adjusted p-value",font=3)
        points(pos[cpg]*10^-6,-0.2,pch=17)
        text(pos[cpg]*10^-6,-0.4,cpgnames[cpg])
        #segments(40.16,3,40.23,3,lwd=2)
        #segments(c(40.16,40.23),2.9,c(40.16,40.23),3.1,lwd=2)
        #text(40.195,2.6,"PPIE",cex=1.2)
        dev.off()
```

```
}


#cg17749961
#gene = LYCAT
#chr2:30670137-30830712

poss <- read.csv("10Cpgs2.csv",as.is=T)
cpgs <- poss[1:11,4]  # this is where the cpg order gets messed up.
These are the 10 closest cpgs to "cg01561916" but have unfortunately
been given the same name as the first bunch of CpGs so when I reran
the code it messed it up
# oldcpgs: "cg24920358" "cg22333868" "cg13926569" "cg17749961"
"cg10106388" "cg06873352" "cg13507326" "cg01561916" "cg14141399"
# [10] "cg18294158"
poss <- poss[1:11,5]

cpg=4
        PPIEgeno <-
read.table(paste(c("cpg",cpg,".ped"),collapse=""),header=F,as.is=T)
        row.names(PPIEgeno) <-
gsub("Tas-","",as.character(PPIEgeno[,1]))
        row.names(PPIEgeno) <- gsub("_",".",row.names(PPIEgeno))
        row.names(PPIEgeno) <- gsub("Tas","",row.names(PPIEgeno))
        #map <- read.table("PPIE.map",as.is=T)
        map <-
read.table(paste(c("cpg",cpg,".map"),collapse=""),as.is=T)
        nsnps <- nrow(map)
        log10pv <- vector(length=nsnps)
        for (i in 1:nsnps) {
                geno <- PPIEgeno[,(i*2+5):(i*2+6)]
                geno <- geno[rowSums(geno=="0")==0,]
                geno <-
geno[match(row.names(Raw_zhang),row.names(geno)),]
                als <- levels(factor(unlist(geno)))
                if (length(als)<=1 |
sum(geno[1]==geno[2],na.rm=T)==0) {
                        next
                }
                gc <- rowSums(geno==als[1])
                fit = lm(Raw_zhang[,cpg] ~ gc)
                pv <- summary(fit)$coef[2,4]
                log10pv[i] <- -1*log10(pv)
        }

png(paste(cpgnames[cpg],"_Raw.png",sep=""),pointsize=12,units="mm",w
idth=137.6,height=137.6*2/3,res=800)
        par(family="serif")
        par(mar=c(4,4,3,0.5))
        plot(map[,4]*10^-6,log10pv,type="o",main="Association
between methylation and SNPs:
        Raw",xlab="Position (Mb)",ylab="-1*log10(p-
value)",xlim=c(map[1,4]*10^-6,map[nsnps,
4]*10^-6+0.4),ylim=c(-1.5,10),cex=0.2)
```

```
        abline(h=-1*log10(0.05),lty=2)
        text(map[nsnps,4]*10^-6+0.2,-1*log10(0.05)+0.5,"p-value =
0.05",font=3)
        abline(h=-1*log10(0.05/nsnps),lty=2)
        text(map[nsnps,4]*10^-6+0.15,-1*log10(0.05/nsnps)
+0.5,"adjusted p-value",font=3)
        points(pos[cpg]*10^-6,-0.2,pch=17)
        text(pos[cpg]*10^-6,-1,cpgnames[cpg])
        #points(as.numeric(poss)*10^-6,rep(-0.2,11),pch=2)
        segments(30.67,9,30.83,9,lwd=2)
        segments(c(30.67,9,30.83),8.7,c(30.67,9,30.83),9.3,lwd=2)
        text(30.83,8,"LYCAT",cex=1)
        dev.off()
# this works for raw now try on other data, ie stratQN and COmBat
#StratQN
for (i in 1:nsnps) {
                geno <- PPIEgeno[,(i*2+5):(i*2+6)]
                geno <- geno[rowSums(geno=="0")==0,]
                geno <-
geno[match(row.names(StratQN_zhang),row.names(geno)),]
                als <- levels(factor(unlist(geno)))
                if (length(als)<=1 |
sum(geno[1]==geno[2],na.rm=T)==0) {
                        next
                }
                gc <- rowSums(geno==als[1])
                fit = lm(StratQN_zhang[,cpg] ~ gc)
                pv <- summary(fit)$coef[2,4]
                log10pv[i] <- -1*log10(pv)
        }

png(paste(cpgnames[cpg],"_StratQN.png",sep=""),pointsize=12,units="m
m",width=137.6,height=137.6*2/3,res=800)
        par(family="serif")
        par(mar=c(4,4,3,0.5))
        plot(map[,4]*10^-6,log10pv,type="o",main="Association
between methylation and SNPs:
        Stratified QN",xlab="Position (Mb)",ylab="-1*log10(p-
value)",xlim=c(map[1,4]*10^-6,map[nsnps,
4]*10^-6+0.4),ylim=c(-1.5,10),cex=0.2)
        abline(h=-1*log10(0.05),lty=2)
        text(map[nsnps,4]*10^-6+0.2,-1*log10(0.05)+0.5,"p-value =
0.05",font=3)
        abline(h=-1*log10(0.05/nsnps),lty=2)
        text(map[nsnps,4]*10^-6+0.15,-1*log10(0.05/nsnps)
+0.5,"adjusted p-value",font=3)
        points(pos[cpg]*10^-6,-0.2,pch=17)
        text(pos[cpg]*10^-6,-1,cpgnames[cpg])
        #points(as.numeric(poss)*10^-6,rep(-0.2,11),pch=2)
        segments(30.67,9,30.83,9,lwd=2)
        segments(c(30.67,9,30.83),8.7,c(30.67,9,30.83),9.3,lwd=2)
        text(30.83,8,"LYCAT",cex=1)
        dev.off()
## There is a big jump in significance!!!
```

```
# Combat_StratQN
for (i in 1:nsnps) {
                geno <- PPIEgeno[,(i*2+5):(i*2+6)]
                geno <- geno[rowSums(geno=="0")==0,]
                geno <-
geno[match(row.names(Combat_StratQN_zhang),row.names(geno)),]
                als <- levels(factor(unlist(geno)))
                if (length(als)<=1 |
sum(geno[1]==geno[2],na.rm=T)==0) {
                        next
                }
                gc <- rowSums(geno==als[1])
                fit = lm(Combat_StratQN_zhang[,cpg] ~ gc)
                pv <- summary(fit)$coef[2,4]
                log10pv[i] <- -1*log10(pv)
        }

png(paste(cpgnames[cpg],"_Combat_StratQN.png",sep=""),pointsize=12,u
nits="mm",width=137.6,height=137.6*2/3,res=800)
        par(family="serif")
        par(mar=c(4,4,3,0.5))
        plot(map[,4]*10^-6,log10pv,type="o",main="Association
between methylation and SNPs:
        Stratified QN with Combat",xlab="Position
(Mb)",ylab="-1*log10(p-value)",xlim=c(map[1,4]*10^-6,map[nsnps,
4]*10^-6+0.4),ylim=c(-1.5,10),cex=0.2)
        abline(h=-1*log10(0.05),lty=2)
        text(map[nsnps,4]*10^-6+0.2,-1*log10(0.05)+0.5,"p-value =
0.05",font=3)
        abline(h=-1*log10(0.05/nsnps),lty=2)
        text(map[nsnps,4]*10^-6+0.15,-1*log10(0.05/nsnps)
+0.5,"adjusted p-value",font=3)
        points(pos[cpg]*10^-6,-0.2,pch=17)
        text(pos[cpg]*10^-6,-1,cpgnames[cpg])
        #points(as.numeric(poss)*10^-6,rep(-0.2,11),pch=2)
        segments(30.67,9,30.83,9,lwd=2)
        segments(c(30.67,9,30.83),8.7,c(30.67,9,30.83),9.3,lwd=2)
        text(30.83,8,"LYCAT",cex=1)
        dev.off()
# this one is in between raw and StratQN, expected as batch and
family are confounded you may actually be taking out real
information

**********************************************************************
# which were the samples used in this analysis?
rownames(Raw_zhang)[row.names(Raw_zhang) %in% row.names(geno)]
 # [1] "PC11.3"   "PC22.2"   "PC11.4"   "PC22.3"   "PC11.9"
"PC22.16"
 # [7] "PC11.147" "PC22.17"  "PC22.21"  "PC22.416" "PC72.4"
"PC9.1"
# [13] "PC9.4"    "PC9.12"   "PC9.477"  "PC22.16"  "PC22.17"
"PC22.17"
# [19] "PC22.4"   "PC11.180" "PC22.17"  "PC9.338"
```

```
samples_both_geno_meth <- rownames(Raw_zhang)[row.names(Raw_zhang)
%in% row.names(geno)]
write.csv(samples_both_geno_meth, file="samples_both_geno_meth.csv")
**********************************************************************


#### Redo plots with KINSHIP and new omni data###
##  Use the association model from the GenABEL package to account
for kinship.  Then plot as above

# take out the replicate samples:  [,-c(27,28,31,41,16)]
Raw_zhang <- Raw_zhang[-c(27,28,31,41,16),]
StratQN_zhang <- StratQN_zhang[-c(27,28,31,41,16),]
Combat_StratQN_zhang <- Combat_StratQN_zhang[-c(27,28,31,41,16),]

# Fix the sample names
rownames(Raw_zhang) <- gsub(".", "_", rownames(Raw_zhang),
fixed=TRUE)
rownames(Raw_zhang) <- gsub("_3", "_03", rownames(Raw_zhang))
rownames(Raw_zhang) <- gsub("_1", "_01", rownames(Raw_zhang))
rownames(Raw_zhang) <- gsub("_2", "_02", rownames(Raw_zhang))
rownames(Raw_zhang) <- gsub("_4", "_04", rownames(Raw_zhang))
rownames(Raw_zhang) <- gsub("_9", "_09", rownames(Raw_zhang))
rownames(Raw_zhang)[c(6:15,17,19,22:28,30:36,38:47)] <- gsub("_0",
"_", rownames(Raw_zhang)[c(6:15,17,19,22:28,30:36,38:47)])
rownames(Raw_zhang)[8]  <- "PC22_17_a" #add 'a' to match genotyping
files
rownames(Raw_zhang)[16] <- "PC72_04_a"

# genotype data
chr <- c(1,12,10,2,1,17,14,2,19,7)
cpg=4
        system(paste(c("/Applications/plink-1.07-mac-intel/plink --
noweb --bfile /Users/ecazaly/Desktop/PhD_Analysis/
Association_2015april/Ass_genetic_input/cutoff_15_final --chr ",
chr[cpg]," --from-bp ",pos[cpg]-10^6," --to-bp ",pos[cpg]+10^6," --
recode --transpose --out /Users/ecazaly/Desktop/PhD_Analysis/
Association_2015april/Ass_genetic_input/cpg",cpg),collapse=""))

poss <- read.csv("/Users/ecazaly/Desktop/PhD_Analysis/R/
NormPaper_2015/10Cpgs2.csv",as.is=T)
cpgs <- poss[1:11,4]
poss <- poss[1:11,5]

library(GenABEL)
convert.snp.tped(tpedfile="Desktop/PhD_Analysis/
Association_2015april/Ass_genetic_input/cpg4.tped",
tfamfile="Desktop/PhD_Analysis/Association_2015april/
Ass_genetic_input/cpg4.tfam", outfile="Desktop/PhD_Analysis/
Association_2015april/Ass_genetic_input/cpg4.raw")
cpg4_tfam=read.table("Ass_genetic_input/cpg4.tfam", header=FALSE)
```

```
# match pheno data to genotype
length(which(cpg4_tfam$V2 %in% rownames(Raw_zhang)))  #39 -- takes
out 8
Raw_zhang_b <- Raw_zhang[c(which(rownames(Raw_zhang) %in% cpg4_tfam
$V2, arr.ind=TRUE)),]
removed <- Raw_zhang[-c(which(rownames(Raw_zhang) %in% cpg4_tfam$V2,
arr.ind=TRUE)),]
rows_removed <- rownames(removed)

#  Create pheno file
ID_Raw_zhang <- matrix(rownames(Raw_zhang))
colnames(ID_Raw_zhang) <- "id"
sex_Raw_zhang <- matrix(champ_data_beta$mset@phenoData$Sex[-
c(27,28,31,41,16)])
colnames(sex_Raw_zhang) <- "sex"
sex_Raw_zhang <- gsub("M", "1", sex_Raw_zhang)
sex_Raw_zhang <- gsub("F", "0", sex_Raw_zhang)
pheno_Raw_zhang <- cbind(ID_Raw_zhang, sex_Raw_zhang, Raw_zhang)
pheno_Raw_zhang_b <- pheno_Raw_zhang[c(which(rownames(Raw_zhang) %in
% cpg4_tfam$V2, arr.ind=TRUE)),]
write.table(pheno_Raw_zhang_b, file="pheno_Raw_zhang.txt")

remove<- which(!(rownames(pheno_Raw_zhang) %in%
rownames(pheno_Raw_zhang_b)))
length(remove)

# Now need to make the genotype file match the samples on the array
famID <- matrix(c(49,33,4, 43, 6, 48, 7, 8, 10, 12, 52, 15, 18, 21,
38, 30, 51, 53, 25, 22, 39, 34, 47, 5, 9, 28, 29, 36, 26, 17, 13,
45, 27, 11, 37, 14, 31, 50, 44))
# check
length(unique(famID)) # 39
Samples_zhang <- cbind(famID, rownames(pheno_Raw_zhang_b))
write.table(Samples_zhang, file="Samples_zhang.txt",
row.names=FALSE, col.names=FALSE, quote=FALSE)

chr <- c(1,12,10,2,1,17,14,2,19,7)
cpg=4
        system(paste(c("/Applications/plink-1.07-mac-intel/plink --
noweb --bfile /Users/ecazaly/Desktop/PhD_Analysis/
Association_2015april/Ass_genetic_input/cutoff_15_final --keep /
Users/ecazaly/Desktop/PhD_Analysis/Association_2015april/
Samples_zhang.txt --chr ", chr[cpg]," --from-bp ", pos[cpg]-10^6,"
--to-bp ",pos[cpg]+10^6," --recode --transpose --out /Users/ecazaly/
Desktop/PhD_Analysis/Association_2015april/Ass_genetic_input/
cpg_b",cpg),collapse=""))

# create new .raw file
convert.snp.tped(tpedfile="Ass_genetic_input/cpg_b4.tped",
tfamfile="Ass_genetic_input/cpg_b4.tfam",
outfile="Ass_genetic_input/cpg_b4.raw")

# create gwaa object
gwaa_cpg4=load.gwaa.data(phenofile="pheno_Raw_zhang.txt",
```

```
genofile="Ass_genetic_input/cpg_b4.raw", force=TRUE)

# create kinship coefficient object
kinship <- read.csv("Ass_genetic_input/kinship_coefficient.csv",
header=TRUE)
# work out which ones I need
Raw_zhang_c <- Raw_zhang_b
rownames(Raw_zhang_c) <- gsub("PC", "PCTAS", rownames(Raw_zhang_c))
rownames(Raw_zhang_c)  <- gsub("_", "-", rownames(Raw_zhang_c) )

kinship$decoded.1 <- gsub("PC22-", "", kinship$decoded.1,
fixed=TRUE)
kinship$decoded.1<- gsub("PC11-", "",kinship$decoded.1, fixed=TRUE)
kinship$decoded.1<- gsub("PC72-", "", kinship$decoded.1, fixed=TRUE)
kinship$decoded.1<- gsub("PC9-", "", kinship$decoded.1, fixed=TRUE)

kinship$decoded.2 <- gsub("PC22-", "", kinship$decoded.2,
fixed=TRUE)
kinship$decoded.2 <- gsub("PC11-", "", kinship$decoded.2,
fixed=TRUE)
kinship$decoded.2 <- gsub("PC72-", "", kinship$decoded.2,
fixed=TRUE)
kinship$decoded.2 <- gsub("PC9-", "", kinship$decoded.2, fixed=TRUE)

length(which(kinship$decoded.1 %in% rownames(Raw_zhang_c)))  #5654
name=rownames(Meth_B_cg13387643_b)
name <- gsub("PC", "PCTAS", name)
name  <- gsub("_", "-", name )
name <- gsub("-a", "", name)
length(which(kinship$decoded.1 %in% name)))

kinship_1_zhang <- kinship[c(kinship$decoded.1 %in%
rownames(Raw_zhang_c), arr.ind=TRUE),]
dim(kinship_1_zhang) # 5655   4  good
kinship_2_zhang <- kinship_1_zhang[c(kinship_1_zhang$decoded.2 %in%
rownames(Raw_zhang_c), arr.ind=TRUE),]
dim(kinship_2_zhang) # 181 4
head(kinship_2_zhang)
kinship_3_zhang <- kinship_2_zhang[-181,c(1,3,4)]
kinship_3_zhang$decoded.1 <- gsub("PCTAS", "PC", kinship_3_zhang
$decoded.1, fixed=TRUE)
kinship_3_zhang$decoded.2 <- gsub("PCTAS", "PC", kinship_3_zhang
$decoded.2, fixed=TRUE)

# make matrix
kinship_4_zhang <- kinship_3_zhang
library(reshape2)

acast(kinship_4_zhang, kinship_4_zhang$decoded.1 ~ kinship_4_zhang
$decoded.2 , value.var=1)
tmp2 <- xtabs(kinship_4_zhang$decoded.1~kinship_4_zhang$decoded.
2+kinship_4_zhang$Kinship.Coefficient, data= kinship_4_zhang)
head(kinship_4_zhang)
```

```
# check kinship by Identity-By-State function ibs() in GenABEL

# Create gwaa object with all SNPs, pulling out subjects of interest
library(GenABEL)
system("/Applications/plink-1.07-mac-intel/plink --noweb --bfile /
Users/ecazaly/Desktop/PhD_Analysis/Association_2015april/
Ass_genetic_input/cutoff_15_final --keep /Users/ecazaly/Desktop/
PhD_Analysis/Association_2015april/Samples_zhang.txt --recode --
transpose --out /Users/ecazaly/Desktop/PhD_Analysis/
Association_2015april/Ass_genetic_input/all")
# After frequency and genotyping pruning, there are 1616835 SNPs
# After filtering, 32 males, 7 females, and 0 of unspecified sex
convert.snp.tped(tpedfile="Ass_genetic_input/all.tped",
tfamfile="Ass_genetic_input/all.tfam", outfile="Ass_genetic_input/
all.raw")
gwaa_all=load.gwaa.data(phenofile="pheno_Raw_zhang.txt",
genofile="Ass_genetic_input/all.raw", force=TRUE)

#Perform IBS
ibs <- ibs(gwaa_all, weight="freq")
colnames(ibs) <- gsub("_", "-", colnames(ibs))
rownames(ibs) <- gsub("_", "-", rownames(ibs))
# if don't specify weight="freq" then the values are off, with some
values like 0.781 and others like 1609264.  Identical samples do not
= 1

# try SNPRelate package
library(SNPRelate)
# Calculate IBD coefficients by KING method of moment.
snpgdsIBDKING {SNPRelate}
ibd <- snpgdsIBDKING()  # looks like a bit of data manipulation
first
head(ibs)

# perform association
library(GenABEL)

# as before, data seemed to need to be altered
gt_cpg4=as.data.frame(as.numeric(gtdata(gwaa_cpg4)))
gwaa_cpg4b <- gwaa_cpg4
gwaa_cpg4b@gtdata@gtps=gt_cpg4
gwaa_cpg4b@phdata$id=gsub("_","-",gwaa_cpg4b@phdata$id)
rownames(gwaa_cpg4b@phdata)=gsub("_", "-",
rownames(gwaa_cpg4b@phdata))
gwaa_cpg4b@gtdata@idnames=gsub("_", "-", gwaa_cpg4b@gtdata@idnames)
rownames(gwaa_cpg4b@gtdata@gtps)=gsub("_", "-",
rownames(gwaa_cpg4b@gtdata@gtps))

modelnum=1
name=vector()
coefs_cpg4=vector()
pvals_cpg4=vector()
log10pvps_cpg4=vector()
for(i in colnames(beta_gwaa_cpg4b@phdata[6])) {
```

```
  for(j in colnames(beta_gwaa_cpg4b@gtdata@gtps)){
         if(sum(beta_gwaa_cpg4b@gtdata@gtps[j],na.rm=T)==0) {
                next
        }
    name[modelnum]= paste(i, j, sep= "/")
    formula_cpg4=as.formula(paste(beta_gwaa_cpg4b@phdata[i], "~",
beta_gwaa_cpg4b@gtdata@gtps[j]))
    association_cpg4=polygenic_hglm(formula_cpg4, ibs,
beta_gwaa_cpg4b)
    coefs_cpg4[modelnum] = summary(association_cpg4$hglm)
$FixCoefMat[2,1]
    pvals_cpg4[modelnum] = summary(association_cpg4$hglm)
$FixCoefMat[2,4]
    log10pvps_cpg4[modelnum]= -1*log10(pvals_cpg4[modelnum])
    modelnum= modelnum + 1
    }
    }
length(which(pvals_cpg4=="NaN"))   #1163
length(pvals_cpg4)                                    #1661
# so using b-values produces 261 less NaN errors
# Good ones: 498  -- how does this compare to the number of snps in
the old 370k window?
length(pvals_cpg4)-length(which(pvals_cpg4=="NaN"))
length(beta_gwaa_cpg4b@gtdata@gtps) - length(pvals_cpg4)
# 46 removed due to all 0s

 # something is still not right because the first few are NA but
worked before..  There is something wrong with the CpG as the loop
works on other CpGs.  Maybe it is the M-val being negative.  Will
try on the B-val
 # Warning messages:
# 1: In log(eigen(Sigma, only.values = TRUE)$values) : NaNs produced
# 2: In pt(abs(as.numeric(FixCoefMat[, 3])), object$dfReFe,
lower.tail = FALSE) :
  # NaNs produced
# 3: In pt(abs(as.numeric(FixCoefMat[, 3])), object$dfReFe,
lower.tail = FALSE) :
  # NaNs produced

#ilogit2()  # from minfi
beta_gwaa_cpg4b <- gwaa_cpg4b
beta_gwaa_cpg4b@phdata$cg17749961 <- ilogit2(beta_gwaa_cpg4b@phdata
$cg17749961)
# this seems to work but still get a warning but gives a value not
NaN
# Warning message:
# In log(eigen(Sigma, only.values = TRUE)$values) : NaNs produced
# I guess this makes sense as this was the first warning msg above
and talks about logs which is what the M-values are.  So looks like
I've sorted the second two and maybe an offset is needed to fix the
log issue

## when the loop runs on all snps the same 3 errors come up
# so it works on 97 but not 98, whats different
```

```
beta_gwaa_cpg4b@gtdata@gtps[,97] #3 x1s
beta_gwaa_cpg4b@gtdata@gtps[,98] # only 2x 1s -- is this below some
MAF??
beta_gwaa_cpg4b@gtdata@gtps[,9]  # this has heaps of values but the
loop dpesn't work..
# the coef for this is negative and small and then the p-val is
NaN..; maybe just need to exclude these ones, see hwo many there are

# otherwise take out log p-val part and see if that changes it
length(which(log10pvps_cpg4=="NaN"))
length(which(pvals_cpg4[1]=="NaN"))
# [1] 1424
length(log10pvps_cpg4)
# [1] 1661
# seems like a large proportion

# the model doesn't like it when a snp is 0 for everyone, it
produces NAs and then an error when working out the p-val and messes
the whole loop up.  The if line says if the column adds up to 0 then
skip the rest of the loop and go onto the next snp.  Must add
na.rm=T as some fo the rows have an NA value
# may need to remove monomorphic alleles, ie those that appear in
only one individual.  see if get an error with this.

# Errors being prodiced:
# Error in GLM.MME(Augy = Augy, AugXZ = AugXZ, starting.delta =
c(b.hat,  :
  # GLM.MME diverged! Try different starting values.
# In addition: Warning message:
# In .coef.trunc(qr, .Call(sparseQR_coef, qr, y), drop = TRUE) :
  # sparseQR_coef(): structurally rank deficient case: possibly
WRONG zeros
# maybe some of the snps come up as all 0s?? try analysis in chunks
to find culprit
# first 10 are ok
# 11:20 ok
# 21:40 ok
# 41:70 ok
# 71:90 ok
# 91:99 error
# 91:95 ok
# 96, 97, 98 ok
# 99 error, yes this is where the whole column is 0.  Could just
mean all are homo, or error.
# remove

# Now PLOT
# log10pvps_cpg4
chr <- c(1,12,10,2,1,17,14,2,19,7)
map <- read.table("Ass_genetic_input/cpg_b4.map", as.is=T)
# cut down map to the 1661 snps that didn't have all 0
name_keep <- gsub("cg17749961/", "", name)
head(map)
map_b <- map[c(which(name_keep %in% map$V2)),]
```

```
nsnps <- nrow(map_b)
pos <-
c(39991652,78121579,89408710,30530779,159099485,59216916,77233176,42
876472,56919728,103657263)

png("cg17749961_Raw_OMNI.png",pointsize=12,units="mm",width=137.6,he
ight=137.6*2/3,res=800)
        par(family="serif")
        par(mar=c(4,4,3,0.5))
        plot(map_b[,4]*10^-6,
log10pvps_cpg4,type="o",main="Association between methylation and
SNPs:
        Raw",xlab="Position (Mb)",ylab="-1*log10(p-
value)",xlim=c(map_b[1,4]*10^-6,map_b[nsnps,
4]*10^-6+0.4),ylim=c(-1.5,10),cex=0.2)
        abline(h=-1*log10(0.05),lty=2)
        text(map_b[nsnps,4]*10^-6+0.2,-1*log10(0.05)+0.5,"p-value =
0.05",font=3)
        abline(h=-1*log10(0.05/nsnps),lty=2)
        text(map_b[nsnps,4]*10^-6+0.15,-1*log10(0.05/nsnps)
+0.5,"adjusted p-value",font=3)
        points(pos[4]*10^-6,-0.2,pch=17)
        text(pos[4]*10^-6,-1,"cg17749961")
        #points(as.numeric(poss)*10^-6,rep(-0.2,11),pch=2)
        segments(30.67,9,30.83,9,lwd=2)
        segments(c(30.67,9,30.83),8.7,c(30.67,9,30.83),9.3,lwd=2)
        text(30.83,8,"LYCAT",cex=1)
        dev.off()

# This it works.  Although it looks like the SNP picked up in the
previous data is njot on this array or had a NaN value because there
are now 2 sig SNPs but they are a fair bit further downstream that
the cpg site.  Which may be good in the fact that the snp would not
be in the probe body unless it's a reverse probe and even then
surely its miles and miles away

# try on the normalised and batch corrected data
# Raw_zhang <- Raw_zhang[-c(27,28,31,41,16),]
# StratQN_zhang <- StratQN_zhang[-c(27,28,31,41,16),]
# Combat_StratQN_zhang <- Combat_StratQN_zhang[-c(27,28,31,41,16),]

rownames(pheno_Raw_zhang_b)
StratQN_zhang_b <- StratQN_zhang
StratQN_zhang_b <- StratQN_zhang_b[-c(remove),]
rownames(StratQN_zhang_b) <- rownames(pheno_Raw_zhang_b)

StratQN_zhang_gwaa <- gwaa_cpg4b
StratQN_zhang_gwaaB <-StratQN_zhang_gwaa
StratQN_zhang_gwaaB@phdata$cg17749961 <-
ilogit2(StratQN_zhang_b[,"cg17749961"])

modelnum=1
name_stratQN=vector()
coefs_stratQN=vector()
```

```
pvals_stratQN=vector()
log10pvps_stratQN=vector()
for(i in colnames(StratQN_zhang_gwaaB@phdata[6])) {
  for(j in colnames(StratQN_zhang_gwaaB@gtdata@gtps)){
          if(sum(StratQN_zhang_gwaaB@gtdata@gtps[j],na.rm=T)==0) {
                  next
          }
    name_stratQN[modelnum]= paste(i, j, sep= "/")
    formula_stratQN=as.formula(paste(StratQN_zhang_gwaaB@phdata[i],
"~", StratQN_zhang_gwaaB@gtdata@gtps[j]))
    association_stratQN=polygenic_hglm(formula_stratQN, ibs,
StratQN_zhang_gwaaB)
    coefs_stratQN[modelnum] = summary(association_stratQN$hglm)
$FixCoefMat[2,1]
    pvals_stratQN[modelnum] = summary(association_stratQN$hglm)
$FixCoefMat[2,4]
    log10pvps_stratQN[modelnum]= -1*log10(pvals_stratQN[modelnum])
    modelnum= modelnum + 1
    }
    }
length(which(pvals_stratQN=="NaN"))   # this is now 0, as opposed to
1163.  Must be the offset etc in Strat
length(pvals_stratQN)                                # all 1661
included

png("cg17749961_StratQN_OMNI.png",pointsize=12,units="mm",width=137.
6,height=137.6*2/3,res=800)
         par(family="serif")
         par(mar=c(4,4,3,0.5))
         plot(map_b[,4]*10^-6,
log10pvps_stratQN,type="o",main="Association between methylation and
SNPs:
         StratQN_OMNI",xlab="Position (Mb)",ylab="-1*log10(p-
value)",xlim=c(map_b[1,4]*10^-6,map_b[nsnps,
4]*10^-6+0.4),ylim=c(-1.5,10),cex=0.2)
         abline(h=-1*log10(0.05),lty=2)
         text(map_b[nsnps,4]*10^-6+0.2,-1*log10(0.05)+0.5,"p-value =
0.05",font=3)

         abline(h=-1*log10(0.05/nsnps),lty=2)
         text(map_b[nsnps,4]*10^-6+0.15,-1*log10(0.05/nsnps)
+0.5,"adjusted p-value",font=3)

         points(pos[4]*10^-6,-0.2,pch=17)
         text(pos[4]*10^-6,-1,"cg17749961")
         #points(as.numeric(poss)*10^-6,rep(-0.2,11),pch=2)
         segments(30.67,9,30.83,9,lwd=2)
         segments(c(30.67,9,30.83),8.7,c(30.67,9,30.83),9.3,lwd=2)
         text(30.83,8,"LYCAT",cex=1)
         dev.off()

Combat_zhang <- Combat_StratQN_zhang
Combat_zhang <- Combat_zhang[-c(remove),]
rownames(Combat_zhang) <- rownames(pheno_Raw_zhang_b)
```

```
Combat_zhang_gwaa  <- gwaa_cpg4b
Combat_zhang_gwaaB <- Combat_zhang_gwaa
Combat_zhang_gwaaB@phdata$cg17749961 <-
ilogit2(Combat_zhang[,"cg17749961"])

         modelnum=1
         name_combat=vector()
         coefs_combat=vector()
         pvals_combat=vector()
         log10pvps_combat=vector()
         for(i in colnames(Combat_zhang_gwaaB@phdata[6])) {
           for(j in colnames(Combat_zhang_gwaaB@gtdata@gtps)){

if(sum(Combat_zhang_gwaaB@gtdata@gtps[j],na.rm=T)==0) {
                         next
                }
             name_combat[modelnum]= paste(i, j, sep= "/")

formula_combat=as.formula(paste(Combat_zhang_gwaaB@phdata[i], "~",
Combat_zhang_gwaaB@gtdata@gtps[j]))
             association_combat=polygenic_hglm(formula_combat, ibs,
Combat_zhang_gwaaB)
             coefs_combat[modelnum] = summary(association_combat
$hglm)$FixCoefMat[2,1]
             pvals_combat[modelnum] = summary(association_combat
$hglm)$FixCoefMat[2,4]
             log10pvps_combat[modelnum]=
-1*log10(pvals_combat[modelnum])
             modelnum= modelnum + 1
             }
             }
length(which(pvals_combat=="NaN"))   # 0
length(pvals_combat)                                    #1661
length(pvals_combat)-length(which(pvals_combat=="NaN")) #1661
length(Combat_zhang_gwaaB@gtdata@gtps) - length(pvals_combat) #46

png("cg17749961_combat_OMNI.png",pointsize=12,units="mm",width=137.6
,height=137.6*2/3,res=800)
         par(family="serif")
         par(mar=c(4,4,3,0.5))
         plot(map_b[,4]*10^-6,
log10pvps_combat,type="o",main="Association between methylation and
SNPs:
         Combat_OMNI",xlab="Position (Mb)",ylab="-1*log10(p-
value)",xlim=c(map_b[1,4]*10^-6,map_b[nsnps,
4]*10^-6+0.4),ylim=c(-1.5,10),cex=0.2)
         abline(h=-1*log10(0.05),lty=2)
         text(map_b[nsnps,4]*10^-6+0.2,-1*log10(0.05)+0.5,"p-value =
0.05",font=3)

         abline(h=-1*log10(0.05/nsnps),lty=2)
         text(map_b[nsnps,4]*10^-6+0.15,-1*log10(0.05/nsnps)
+0.5,"adjusted p-value",font=3)
```

```
        points(pos[4]*10^-6,-0.2,pch=17)
        text(pos[4]*10^-6,-1,"cg17749961")
        #points(as.numeric(poss)*10^-6,rep(-0.2,11),pch=2)
        segments(30.67,9,30.83,9,lwd=2)
        segments(c(30.67,9,30.83),8.7,c(30.67,9,30.83),9.3,lwd=2)
        text(30.83,8,"LYCAT",cex=1)
        dev.off()

# also try with the original model with the new OMNI data
# for (i in 1:nsnps) {
                # geno <- PPIEgeno[,(i*2+5):(i*2+6)]
                # geno <- geno[rowSums(geno=="0")==0,]
                # geno <-
geno[match(row.names(StratQN_zhang),row.names(geno)),]
                # als <- levels(factor(unlist(geno)))
                # if (length(als)<=1 |
sum(geno[1]==geno[2],na.rm=T)==0) {
                        # next
                # }
                # gc <- rowSums(geno==als[1])
                # fit = lm(StratQN_zhang[,cpg] ~ gc)
                # pv <- summary(fit)$coef[2,4]
                # log10pv[i] <- -1*log10(pv)
        # }
  for(i in 1:length(Combat_zhang_gwaaB@gtdata@gtps)){ i=1
         if(sum(gwaa10@gtdata@gtps[i],na.rm=T)==0) {
                next
                }
                fit = lm(gwaa10@phdata[,6] ~
gwaa10@gtdata@gtps[,i])
                pv <- summary(fit)$coef[2,4]
                log10pv[i] <- -1*log10(pv)
        }
length(log10pv)  # this is giving a pval length of 64779

chr <- c(1,12,10,2,1,17,14,2,19,7)
map <- read.table("Ass_genetic_input/cpg_b4.map", as.is=T)
# cut down map to the 1661 snps that didn't have all 0
name_keep <- gsub("cg17749961/", "", name)
head(map)
map_b <- map[c(which(name_keep %in% map$V2)),]
nsnps <- nrow(map_b)
pos <-
c(39991652,78121579,89408710,30530779,159099485,59216916,77233176,42
876472,56919728,103657263)

png("test.png",pointsize=12,units="mm",width=137.6,height=137.6*2/3,
res=800)
        par(family="serif")
        par(mar=c(4,4,3,0.5))
        plot(map_b[,4]*10^-6, log10pv,type="o",main="Association
between methylation and SNPs:
        Combat_OMNI_oldMODEL",xlab="Position
(Mb)",ylab="-1*log10(p-value)",xlim=c(map_b[1,4]*10^-6,map_b[nsnps,
```

```
4]*10^-6+0.4),ylim=c(-1.5,10),cex=0.2)
        abline(h=-1*log10(0.05),lty=2)
        text(map_b[nsnps,4]*10^-6+0.2,-1*log10(0.05)+0.5,"p-value =
0.05",font=3)
        abline(h=-1*log10(0.05/nsnps),lty=2)
        text(map_b[nsnps,4]*10^-6+0.15,-1*log10(0.05/nsnps)
+0.5,"adjusted p-value",font=3)
        points(pos[4]*10^-6,-0.2,pch=17)
        text(pos[4]*10^-6,-1,"cg17749961")
        #points(as.numeric(poss)*10^-6,rep(-0.2,11),pch=2)
        segments(30.67,9,30.83,9,lwd=2)
        segments(c(30.67,9,30.83),8.7,c(30.67,9,30.83),9.3,lwd=2)
        text(30.83,8,"LYCAT",cex=1)
        dev.off()

# Use the kinship calculated for a subset of samples
samples <- c('PC9-1 (AH1)',"PC9-4 (AH2)","PC9-12 (AH4)", "PC9-24
(AH6)", "PC9-121 (AH7)", "PC9-477 (AH8)", "PC22-16 (AH9)", "PC22-17
(AH10)", "PC22-210 (AH11)", "PC22-393 (AH12)", "pc11-3", "pc22-2",
"pc11-4", "pc22-3", "pc11-9", "pc22-16", "pc11-147", "pc22-17",
"pc22-21", "PC22-468", "pc22-203", "pc22-387", "pc72-136",
"pc22-416", "pc72-188", "pc72-4", "pc72-77", "pc72-126")
# have removed # 26 and 28 not present pc72-213, pc72-213

Raw_d <- getM(champ_data_mset_corrected)
colnames(Raw_d) <- champ_data_beta$mset@phenoData$Sample_Name
Raw_d2 <- Raw_d[,samples]
meth_dist_raw <- as.matrix(dist(t(Raw_d2)))
dimnames(meth_dist_raw)=list(colnames(Raw_d2), colnames(Raw_d2))
write.csv(meth_dist_raw, file="meth_dist_raw.csv") # use the
distances in matrix to fill in kinship vs meth excel table, then
reload

StratQN_d_M <- getM(champ_StratQN)
colnames(StratQN_d_M) <- champ_data_beta$mset@phenoData$Sample_Name
StratQN_d_M <- StratQN_d_M[,samples]
meth_dist_StratQN_d_M <- as.matrix(dist(t(StratQN_d_M)))
write.csv(meth_dist_StratQN_d_M, file="meth_dist_StratQN.csv")

StratQNcombat_d_M <- as.matrix(Combat_StratQN_corrected_M)
colnames(StratQNcombat_d_M) <-champ_data_beta$mset@phenoData
$Sample_Name
StratQNcombat_d_M <- StratQNcombat_d_M[,samples]
meth_dist_StratQNcombat <- as.matrix(dist(t(StratQNcombat_d_M)))
write.csv(meth_dist_StratQNcombat,
file="meth_dist_StratQNcombat.csv")

meth_dist_kinship <- read.csv('NormPaper/Kinship vs
Methdis_Mar15.csv')

par(mfrow=c(1,3))
plot(meth_dist_kinship$Meth.Dist.Raw..Mval~meth_dist_kinship
$Kinship..Coefficient, col=meth_dist_kinship$Family, pch=16,
ylim=c(200,1000), ylab='Methylation Distance: M-vals', xlab='Kinship
```

```
Coefficient (absolute Log)', main="Raw Data
Methylation Distance vs Kinship Coefficient")
legend('topleft', col=1:3, pch=16, legend=levels(meth_dist_kinship
$Family))
plot(meth_dist_kinship$Meth.Dist.StratQN..Mval~meth_dist_kinship
$Kinship..Coefficient, col=meth_dist_kinship$Family, pch=16,
ylim=c(200,1000), ylab='Methylation Distance: M-vals', xlab='Kinship
Coefficient (absolute Log)', main="Stratified QN
Methylation Distance vs Kinship Coefficient")
legend('topleft', col=1:3, pch=16, legend=levels(meth_dist_kinship
$Family))
plot(meth_dist_kinship
$Meth.Dist.Combat.StratQN..Mval.~meth_dist_kinship
$Kinship..Coefficient, col=meth_dist_kinship$Family, pch=16,
ylim=c(200,1000), ylab='Methylation Distance: M-vals', xlab='Kinship
Coefficient (absolute Log)', main="Stratified QN, ComBat corrected
Methylation Distance vs Kinship Coefficient")
legend('topleft', col=1:3, pch=16, legend=levels(meth_dist_kinship
$Family))

plot(meth_dist_kinship
$Meth.Dist.Combat.StratQN..Mval.~meth_dist_kinship
$Kinship..Coefficient, col=meth_dist_kinship$Family,
pch=meth_dist_kinship$Individual.1, ylim=c(200,400),
ylab='Methylation Distance: M-vals', xlab='Kinship Coefficient
(absolute Log)', main="Stratified QN, ComBat corrected
Methylation Distance vs Kinship Coefficient")
legend('topleft', col=1:3, pch=16, legend=levels(meth_dist_kinship
$Family))


# Perhaps just look at those closely related as the relationship
tends to drop off after that

close_kinship <- meth_dist_kinship[meth_dist_kinship
$Kinship..Coefficient <1.0,]
Family_coef <- close_kinship$Family
Family_coef <- gsub("Family 9", "3", Family_coef)
Family_coef <- gsub("Family 22", "1", Family_coef)
Family_coef <- gsub("Family 11", "2", Family_coef)

par(mfrow=c(1,3))
plot(close_kinship$Meth.Dist.Raw..Mval~close_kinship
$Kinship..Coefficient, cex=1.5, pch=Family_coef, col=Family_coef,
ylim=c(200,810), ylab='Methylation Distance: M-vals', xlab='Kinship
Coefficient (absolute Log)', main="Methylation Distance vs Kinship
Coefficient:
 Raw Data")
plot(close_kinship$Meth.Dist.StratQN..Mval~close_kinship
$Kinship..Coefficient, cex=1.5,pch=Family_coef, col=Family_coef,
ylim=c(200,810), ylab='Methylation Distance: M-vals', xlab='Kinship
Coefficient (absolute Log)', main="Methylation Distance vs Kinship
Coefficient:
Stratified QN")
```

```
plot(close_kinship$Meth.Dist.Combat.StratQN..Mval.~close_kinship
$Kinship..Coefficient, cex=1.5, pch=Family_coef, col=Family_coef,
ylim=c(200,810), ylab='Methylation Distance: M-vals', xlab='Kinship
Coefficient (absolute Log)', main="Methylation Distance vs Kinship
Coefficient:
Stratified QN, ComBat corrected")

correlation_Raw <- cor(x=close_kinship$Meth.Dist.Raw..Mval,
y=close_kinship$Kinship..Coefficient, method="pearson")
# -0.6517868
correlation_StratQN <- cor(x=close_kinship$Meth.Dist.StratQN..Mval,
y=close_kinship$Kinship..Coefficient, method="pearson")
# -0.3562952
correlation_Combat_StratQN <- cor(x=close_kinship
$Meth.Dist.Combat.StratQN..Mval., y=close_kinship
$Kinship..Coefficient, method="pearson")
# 0.2796014



#### Positive Control 2 :  QQ plots ####
# creat 3 QQ plots looking at methylation vs age, before and after
normalisation and with Combat

# Run a regression between methylation and age and pull out the p-
value for methylation at a given row.

## For Raw methylation values  ##
rawmeth_age_pv <- function(row) {
  summary(lm(Raw_manual_M_3[row,]~Age_numeric))$coef[2,4]
}
# Test first 100 rows:
pvals <- sapply(1:100,rawmeth_age_pv)
# Obtain p-values for all
pvals_raw <- sapply(1:nrow(Raw_manual_M_3),rawmeth_age_pv)

## For StratQN ##
StratQN_age_pv <- function(row) {
  summary(lm(StratQN_M_2[row,]~Age_numeric))$coef[2,4]
}
# Test first 100 rows:
pvals_StratQN_test <- sapply(1:100,StratQN_age_pv)
# Obtain p-values for all
pvals_StratQN <- sapply(1:nrow(StratQN_M_2), StratQN_age_pv)

## For Combat/StratQN ##
CombatStratQN_age_pv <- function(row) {
  summary(lm(Combat_StratQN_corrected_M[row,]~Age_numeric))
$coef[2,4]
}
# Test first 100 rows:
pvals_CombatStratQN_test <- sapply(1:100, CombatStratQN_age_pv)
# Obtain p-values for all
pvals_CombatStratQN <- sapply(1:nrow(Combat_StratQN_corrected_M),
```

```
CombatStratQN_age_pv)


#### Then generate expected pvals for each  ####
m=length(pvals_raw) #all 3 have same length
expect.stats=-log10(seq(1/(m+1),m/(m+1),length.out=m))

png("QQplot Raw.png")
qqplot(x=expect.stats, y=-log10(pvals_raw), plot.it=TRUE, xlab =
"expected -log10(pvalues)", ylab ="observed -log10(pvalues)",
main="Raw Q-Q Plot", ylim=c(0,14))
abline(a=0,b=1,lwd=2)
lambda_raw=median(-log10(pvals_raw))/median(expect.stats)
text(0,10,bquote(lambda==.(round(lambda_raw,3))),adj=0,cex=2)
dev.off()

png("QQplot StratQN.png")
qqplot(x=expect.stats, y=-log10(pvals_StratQN), plot.it=TRUE, xlab
="expected -log10(pvalues)", ylab ="observed -log10(pvalues)",
main="StratQN Q-Q Plot", ylim=c(0,14))
abline(a=0,b=1,lwd=2)
lambda_strat=median(-log10(pvals_StratQN))/median(expect.stats)
text(0,10,bquote(lambda==.(round(lambda_strat,3))),adj=0,cex=2)
dev.off()

png("QQplot Combat.png")
qqplot(x=expect.stats, y=-log10(pvals_CombatStratQN), plot.it=TRUE,
xlab = "expected -log10(pvalues)", ylab = "observed -
log10(pvalues)", main="Combat Q-Q Plot", ylim=c(0,14))
abline(a=0,b=1,lwd=2)
lambda_combat=median(-log10(pvals_CombatStratQN))/
median(expect.stats)
text(0,10,bquote(lambda==.(round(lambda_combat,3))),adj=0,cex=2)
dev.off()

# these plots show that after normalisation there are a lot more
sites significantly associated with age.
```

Cazaly et al. Clinical Epigenetics (2016) 8:75
DOI 10.1186/s13148-016-0241-2

Clinical Epigenetics

METHODOLOGY Open Access

Comparison of pre-processing methodologies for Illumina 450k methylation array data in familial analyses

Emma Cazaly[1], Russell Thomson[1,2], James R. Marthick[1], Adele F. Holloway[3], Jac Charlesworth[1] and Joanne L. Dickinson[1*]

Abstract

Background: Human methylome mapping in health and disease states has largely relied on Illumina Human Methylation 450k array (450k array) technology. Accompanying this has been the necessary evolution of analysis pipelines to facilitate data processing. The majority of these pipelines, however, cater for experimental designs where matched 'controls' or 'normal' samples are available. Experimental designs where no appropriate 'reference' exists remain challenging. Herein, we use data generated from our study of the inheritance of methylome profiles in families to evaluate the performance of eight normalisation pre-processing methods. Fifty individual samples representing four families were interrogated on five 450k array BeadChips. Eight normalisation methods were tested using qualitative and quantitative metrics, to assess efficacy and suitability.

Results: Stratified quantile normalisation combined with ComBat were consistently found to be the most appropriate when assessed using density, MDS and cluster plots. This was supported quantitatively by ANOVA on the first principal component where the effect of batch dropped from $p < 0.01$ to $p = 0.97$ after stratified QN and ComBat. Median absolute differences between replicated samples were the lowest after stratified QN and ComBat as were the standard error measures on known imprinted regions. Biological information was preserved after normalisation as indicated by the maintenance of a significant association between a known mQTL and methylation ($p = 1.05e\text{-}05$).

Conclusions: A strategy combining stratified QN with ComBat is appropriate for use in the analyses when no reference sample is available but preservation of biological variation is paramount. There is great potential for use of 450k array data to further our understanding of the methylome in a variety of similar settings. Such advances will be reliant on the determination of appropriate methodologies for processing these data such as established here.

Keywords: Familial data, 450k, Array, Methylation, Pre-processing pipeline, Normalisation

Background

DNA methylation, the covalent addition of a methyl group to a cytosine base, usually in a cytosine-guanine pair (CpG), remains the most widely studied epigenetic modification in disease. While around 70 % of CpG dinucleotides are methylated in mammals, when clustered in groups or 'islands' (CGIs) they are generally unmethylated [1]. These islands occur often at promoter regions, where methylation has been traditionally associated with transcriptional repression [2]. Less extensively studied, but potentially more interesting, is the regulatory role of methylation at CpG shores and within gene bodies, as these regions have been found to be more variably methylated between tissue types and in cancer compared to normal tissue [3, 4].

Deepening the complexities surrounding the regulatory roles of CpG dinucleotides located in regions adjacent to promoters, 'shores' and gene bodies is the knowledge that sequence variation has a strong influence on methylation. Gertz et al. [5] examined methylation patterns in a three generation family and have estimated that genotype

* Correspondence: jo.dickinson@utas.edu.au
[1]Menzies Institute for Medical Research, University of Tasmania, Private Bag 23 Medical Sciences Building 2, Hobart, TAS, Australia
Full list of author information is available at the end of the article

explains around 80 % of the variation in methylation. Methylation quantitative trait loci or meQTLs refer to sequence variants across the genome driving methylation patterns [6] and these have been mapped in a variety of different tissues and at different stages of development in various organisms [7–10]. Smith et al. [9] have compared sequence variants influencing methylation patterns across different human tissues and identified sets of meQTLs that are tissue specific but also others that are consistent across different tissue types and indeed across populations. Further, inherited genetic variants have been linked to methylation changes observed in disease. Shen et al. [11] have demonstrated that susceptibility SNPs at the *HNF1B* locus in ovarian cancer are associated with altered methylation and consequent expression of *HNF1B*. Also, it has been proposed that at least a proportion of unexplained Lynch syndrome cases are likely to be due to epigenetic silencing of mismatch repair genes. Indeed, it has been shown that the inheritance of the c.-27C>A germ-line variant in the 5′ UTR leads to epigenetic silencing *MLH1* in Lynch syndrome [12]. Thus, there is now considerable interest in mapping inherited methylation changes influencing disease susceptibility and disease course.

Genome-wide epigenetic studies have thus far largely focused on epigenetic alterations that occur in diseased tissues, where epigenetic changes across the genome are mapped through comparing 'normal' and affected tissues from the same individual. Indeed, epigenetic drugs, currently in clinical use, are designed to correct the epigenetic alterations acquired during disease development [13]. The assumption being that these acquired epigenetic alterations are driven by the disease process itself. More recently, it has been hypothesised that inherited genetic variation can drive epigenetic alterations and further that these contribute to disease susceptibility or disease course. To date, the large majority of genome-wide methylation studies and consequently the bioinformatic pipelines used to interpret these data have been designed to compare diseased with 'normal' tissue, in order to map epigenetic changes in the disease tissue itself. This analysis may screen out inherited epigenetic changes that are evident both in the normal tissue and the diseased tissue of the same affected individual. There remains a need to explore inter-individual variation of the epigenome and its contribution to disease. A powerful approach to examining the role of inherited variation drivers of epigenetic change is to examine large families where inheritance of variation driving epigenetic alterations can be tracked through generations.

A number of challenges exist in the analysis of genome-wide methylation mapping in samples and these include technical challenges dealing with batch effects and the underlying biochemistry employed by the array methods. This has necessitated the development of numerous pre-processing quality control methods to ensure reliable, high-quality data generation. As most studies examining epigenetic profiles have typically examined differences between two distinct groups (normal vs tumour tissue or case vs control), the majority of normalisation methods for the 450k array are designed for these types of data, frequently requiring two data groups to normalise negative and positive control probes or genomic regions. Such methods are incompatible with pedigree data, which lack a distinct second group for normalisation. In response to the absence of appropriate strategies, we have developed a pipeline for optimal normalisation and pre-processing of familial-based methylation array data.

Methods

DNA isolation and preparation

Fifty peripheral blood samples were collected from individuals representing clusters of densely aggregated cases of affected men and close relatives from the Tasmanian Familial Prostate Cancer study. A diagrammatic representation of the family pedigrees is shown in Fig. 1, with disease status indicated. Individuals are of Caucasian descent, ranging in age from 23 to 89 years. See Additional file 1: Table S1 for more detailed information on clinical data and sample handling where available. DNA was extracted from whole blood using the Nucleon BACC3 (GE Healthcare) DNA extraction kit, following the manufacturer's instructions. DNA was initially quantified on the Nanodrop 8000 (Thermo Scientific) and samples with a 260:280 ratio of less than 1.80 were further purified using the Zymo Clean & Concentrator (TM)-5 Kit. DNA was then quantified using a Qubit® Flourometer. One microgram of DNA was bisulphite converted using the EZ DNA Methylation-Gold (TM) kit (ZYmo Research), according to the manufacturer's instructions. Bisulphite-converted DNA (400 ng) was then used for analysis of DNA methylation using the 450k array, according to the manufacturer's instructions.

Data extraction, pre-processing and initial quality control

IDAT files containing the raw intensity signals from red and green colour channels were generated using Illumina's *iControl* software, with all further analysis carried out in the R environment [14]. A combination of three R packages, *minfi* [15], *methylumi* [16] and *ChAMP* [17], were used to load *IDAT* files into R and perform basic quality control. Different normalisation methods require the data to be in different formats which cannot be subsequently modified once loaded into *R*. As such, a number of different packages were used to load data, with the chosen package dependent on the normalisation method tested. *Methylumi* was used to read data into R in the correct format for quantile normalisation in the *lumi R* package. The *minfi* package provides a quality control report based on inbuilt control probes on the

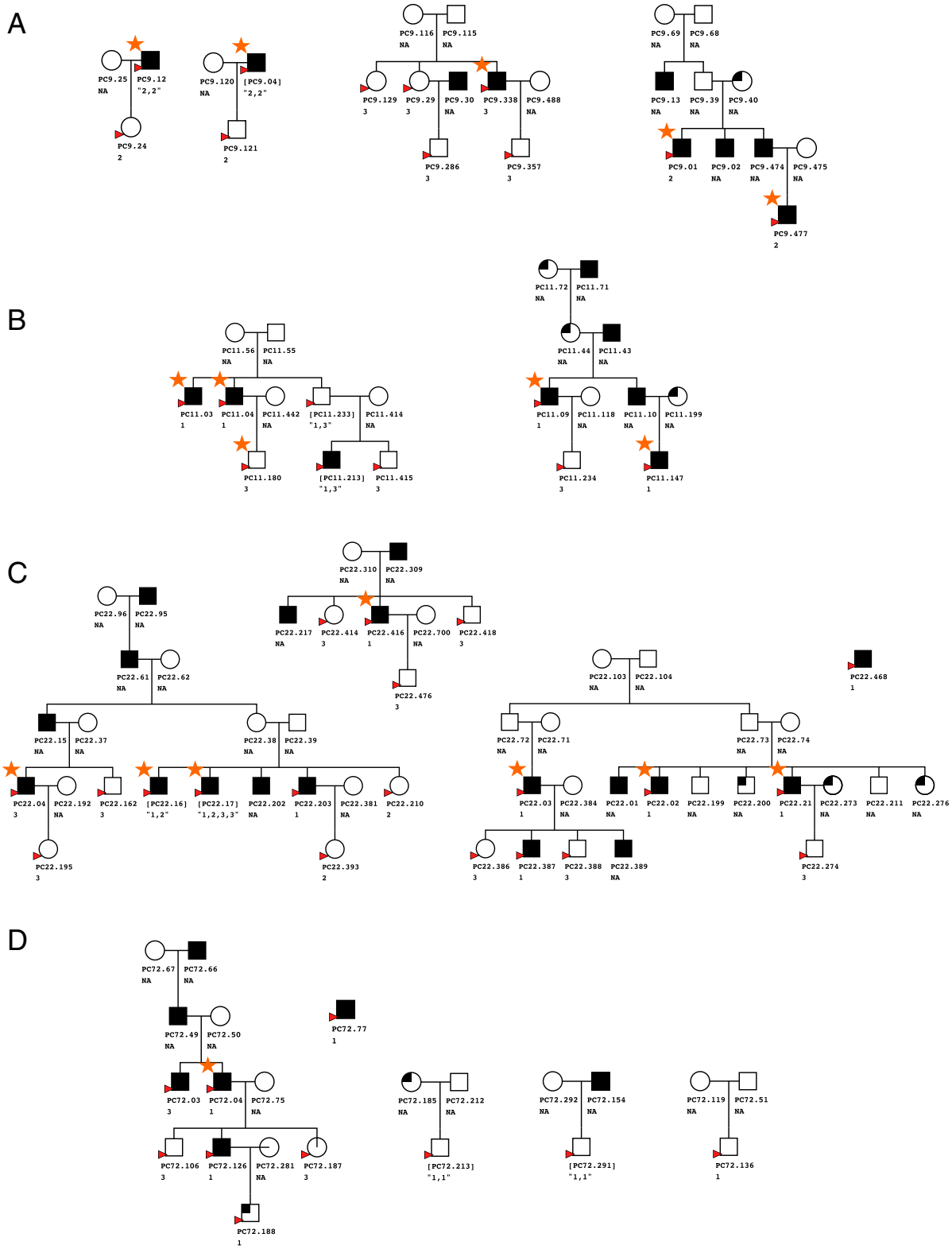


Fig. 1 (See legend on next page.)

(See figure on previous page.)
Fig. 1 Selected pedigree clusters from four families from the Tasmanian Familial Prostate Cancer study. Four clusters were chosen from family 9 (**a**), two from family 11 (**b**), four from family 22 (**c**) and five from family 72 (**d**). *Circles* represent women and *squares* men, with individuals affected by prostate cancer filled in *black*, those unaffected unfilled and individuals affected by other cancers quarter filled. Samples interrogated on the 450k array are indicated by a *red arrow head*. Replicate samples are indicated by *square brackets* around the sample name, while the batch is indicated underneath the sample name. *Orange stars* indicate samples for which good-quality Omni2.5 genotype and 450k methylation data were available

array (such as staining, hybridization, bisulfite conversion and negative controls) as well as the ability to exclude probes and samples based on probe signal intensity. Samples failing this initial quality control were excluded from further analysis. Replicate samples across batches were included on the beadchips to allow assessment of quality control and technical bias. Of the 50 unique samples and 8 replicates initially interrogated, 45 unique and 5 replicate samples passed quality control metrics and were used for further analysis. Following sample quality control, the recommended quality thresholds in ChAMP were employed to exclude poor quality probes, with a minimum detection p value of 0.05 in more than one sample removing 6740 probes and a bead count threshold of <3 in 5 % of samples removing a further 478 probes. To account for sex differences in methylation, driven particularly by dosage compensation by X-inactivation, probes on the sex chromosomes were removed prior to normalisation. While ChAMP includes this option as default when loading data, most packages require manual separation, normalisation and recombination of sex chromosomes or their complete manual removal. Thus, to permit appropriate comparison of normalisation methods, a homogenous set of loci across all packages was required; therefore, sex chromosomes were removed at this stage of analysis and not re-introduced.

Normalisation

Eight normalisation techniques were applied to the whole dataset, as detailed in Table 1 with each method evaluating the same samples. The probe subset chosen for each analysis was selected following the instructions of each individual normalisation package, which had different requirements. This dictated whether normalisation methods were compatible and could be used in conjunction.

Data are presented for each method except RUV, for which the results were not resolvable using the data generated in this study. These methods involve various degrees of type I and II probe scaling to account for underlying technical differences between the probe types, background and dye bias correction and initial between array batch correction. Depending on the normalisation method, data was either used in the red/green signal format (RGset), converted into methylated and unmethylated values (MethylSet) or converted to β values by the function $\beta = M/(M + U + 100)$, where M is the methylated

Table 1 Normalisation methods tested. The table includes a brief description of each method, the relevant R package and reference for further information

Normalisation method	Package	Reference
Quantile normalisation The distributions of probe intensities for different samples are made identical. Often used in microarray analysis.	lumi	[33]
Stratified quantile normalisation Probes are stratified by genomic region then quantile normalised with sex chromosomes normalised separately when male and female samples are present. No background correction, zeros removed by outlier function. Not recommended for cancer-normal comparisons or other groups with global differences.	minfi	[15]
Beta-mixture quantile dilation (BMIQ) Adjusts type II probes to type I distribution. Recommended for all datasets.	ChAMP	[27]
Subset-quantile within array normalisation (SWAN) A quantile distribution is created using a subset of probes, with subsetting based on the number of CpGs in the probe body. Separate subsets are created for type I and II probes. The remaining probes are then adjusted to the subsets.	minfi	[34]
Functional normalisation (FunNorm) Uses control probes to remove unwanted technical variation. Also diminishes batch effects in some datasets. Suitable for use in cancer-normal studies or where global methylation changes occur.	minfi	[29]
Dasen Background adjustment and between array normalisation are performed separately on type I and II probes.	wateRmelon	[20]
Noob Uses type I probe design to measure non-specific fluorescence in the opposite colour channel.	minfi	[35]
Remove unwanted variation (RUV) Previously used with microarray data to normalise via negative control genes. Requires distinct groups such as cancer-normal to normalise on.	RUVnormalize	[36]
Batch correction: *ComBat* Adjusts for known or unknown batches using an empirical Bayesian framework.	sva	[19]

signal and U unmethylated. In some normalisation methods, the offset of 100 is included to regularise scores when both methylated and unmethylated values are very low. While the β value is more biologically intuitive (it ranges from 0 to 1 indicating the proportion of methylation at that site for the population of cells analysed), it suffers from severe heteroskedasticity at very high or low values [18]. Logit transforming to an M value removes this unequal variance. Thus wherever possible, calculations in this study have been performed on the M values and transformed back to β values if required for biological interpretation. Eight performance metrics were then used to compare methods and determine the optimal normalisation approach for familial datasets. Visual tools such as density and MDS plots and unsupervised hierarchical clustering were used to compare the various methods between all samples and particularly replicate samples. See Table 2 for a description of each metric.

Table 2 Qualitative and Quantitative metrics used to assess normalisation efficacy. The table includes a brief description of each metric and which figures describe the results for that method

	Method	Description	Figure
1	Density plot: all samples	Bimodal distribution of Beta values as methylated and unmethylated signals. Each sample is represented by a single line. A batch effect is indicated when samples performed in the same batch have a similar distribution.	Fig. 2a, c, e Additional file 5: Figure S4
	Density plot: three groups of replicate samples	Bimodal distribution of Beta values as methylated and unmethylated signals. Samples are coloured by replicate group. As each replicate group contain the same biological information, differences in sample distribution within groups indicate technical bias.	Additional file 3: Figure S2 (A, C, E)
	Density plot: probe I and II distribution	Bimodal distribution of Beta values as methylated and unmethylated signals separated by Infinium I and II probe types. Provides information about probe normalisation which is required for Infinium I and II signals to be combined in the same analysis.	Fig. 2b, d, f
2	MDS plot: all samples	Multidimensional scaling plots show a 2D projection of distances between samples. For these plots the 1000 most variable sites have been selected as they are the most biologically relevant for this type of analysis. Samples cluster by similarity and as such batch effects and familial clustering can be clearly discerned.	Fig. 3 Additional file 8: Figure S5
	MDS plot: three groups of replicate samples	1000 most variable sites are again selected, with samples coloured by replicate group. As each replicate group contains the same biological information, close within group clustering indicates minimal technical bias while distantly clustered replicate samples indicate heightened technical bias.	Additional file 3: Figure S2 (B, D, F)
3	ANOVA of the first principal component for MDS plots	Provides a quantitative value for MDS plots. A lower p value indicates the clustering is more significantly explained by batch. Ie. a larger p value after normalisation indicates a reduction in batch effect.	p values displayed on Fig. 3
4	Median absolute differences between replicate samples	For each replicate group the median M value (log of Beta values) across all probes was calculated and the absolute difference compared between replicate groups after various normalisation methods. A smaller absolute difference indicates improved normalisation as more technical bias is removed.	Additional file 6: Table S2
5	Imprinted regions: density plots	227 probes mapping known imprinted hemi-methylated regions can be used as a standard to measure changes in methylation levels after normalisation. Density plots have a single distribution peak since there is roughly 50 % methylation at these sites.	Additional file 4: Figure S3
	Differentially methylated region standard error (DMRSE)	The DMRSE measures how each sample varies from the expected 50 % methylation. Smaller error/deviation from 50 % indicates less technical bias.	Additional file 1: Table S1 Additional file 4: Figure S3 (A, C, E)
6	Cluster dendrogram	Another tool to measure clustering by sample similarity. Samples are labelled by batch with batch effects clearly seen before normalisation and diminished after. Red stars indicate replicate samples that are expected to cluster most closely.	Additional file 2: Figure S1
7	meQTL association	Association between methylation at cg17749961 and SNPs in a 2-Mb window. A significant association is maintained after normalisation and batch correction.	Additional file 5: Figure S4
8	Epigenome-wide methylation association with age	QQ plots depicting the association between epigenome-wide methylation and age. Plots are performed on raw, normalised and batch-corrected data.	Additional file 9: Figure S6

Batch correction

Since an obvious batch effect remained after normalisation, the ComBat function from the *sva* package [19] was used to further remove technical bias introduced by interrogating samples on the 450k array in different batches.

Genotype data

DNA from a subset of samples was extracted as described above and interrogated on Illumina's HumanOmni2.5-8 Beadchip according to the manufacturer's instructions. Quality control was performed with Illumina's *GenomeStudio* Software.

Statistical analysis

Eight methods, as described in Table 2, were used to compare the efficacy of the various normalisation methods. In addition to density and MDS plots, the ANOVA test and quantitative measures, mean absolute difference between replicates and the differentially methylated region standard error (DMRSE) measures were used. Additionally, two approaches were taken to test the underlying biological information was preserved between samples; namely, an association analysis between genotype and methylation at a previously identified meQTL and an epigenome-wide association analysis with age.

For a qualitative measure to examine effectiveness of between array normalisation, hierarchical cluster dendrograms were generated using all probes with the *hclust* function using the Euclidean distance between from the default R package, *stats*. Cluster dendrograms group samples by differences, with similar samples grouping together.

MDS plots were clustered by batch or family; then, analysis of variance was performed on the first principal component from a PCA on the 1000 most variable beta values using the *aov* and *prcomp* functions in the *stats* core R package. p values are displayed on the MDS plots in Fig. 2. A lower p value indicates that clustering is more significantly explained by batch or family, with a larger p value after normalisation indicating a reduction in technical bias.

Six replicate sample pairs were used to quantitatively assess the performance of the normalisation methods, as one sample from each pair was interrogated on a separate batch. The median absolute difference between each pair was calculated by first taking the absolute difference at each probe between the two replicates and then taking the median of the differences. A lower median difference indicates less technical bias, as the samples are biologically identical.

There are 227 known imprinted regions (iDMRs) on the 450k array, and these have previously been employed in analysis packages such as *wateRmelon* as a quality control metric [20]. These regions are expected to have allele-specific methylation and a β value of 0.5, and therefore deviation from this value can be examined as a standard error-type measure, denoted DMRSE in the *wateRmelon* package. The *dmrse_row* function was used to measure dispersion of methylation between samples for each normalisation method. A lower value indicates methylation values are more tightly aligned with expected methylation levels.

While evidence of clustering according to familial relationships following normalisation correction provides some confidence that biological integrity of the data is preserved, to further test the preservation of biologically relevant information, we examined detectable associations of known meQTLs in our data. Shoemaker and colleagues have previously identified 736 CpG sites to be associated with SNPs in *cis* [21]. Here, we examined cg17749961, one of the ten most significant hits reported by Shoemaker et al., in the 22 individuals, for whom both methylation and genotyping SNP data was available. Association analysis was performed between this probe site and SNPs located within a 2-Mb window adjacent to this site, using linear regression, and assuming an additive disease model. Relatedness was adjusted for by fitting a linear mixed model on the methylation of cg17749961 and a kinship matrix, determined by the identity-by-state function in the *GenABEL* R package [22]. The residuals from this model were then used as the outcome variable in the linear regression model with SNPs drawn from a 370 K Illumina array. Bonferroni correction was used to correct for multiple testing error.

To further demonstrate biological information is preserved after normalisation and batch correction, the association between age and epigenome-wide methylation was compared for raw data, stratified QN normalised data and ComBat-corrected stratified QN data. Linear regression models were fitted with age as the explanatory variable and methylation as the outcome variable, with −log10 p values of the models plotted against −log10 expected p values as QQ plots.

Results

Evaluation of normalisation methods to address technical bias

Data generated from whole genome methylation analysis employing array technology generates an output necessitating application of normalisation methods to correct for possible bias arising from within and between array variation. Herein eight different methodologies (Table 1) were examined and visual and quantitative metrics were employed to evaluate their comparative performance. High-quality methylation data was generated for 45 unique and five replicate samples from four families using the 450k array in three separate batches (see Fig. 1 for further details). A minimum of one sample in each of the three batches was replicated, providing five

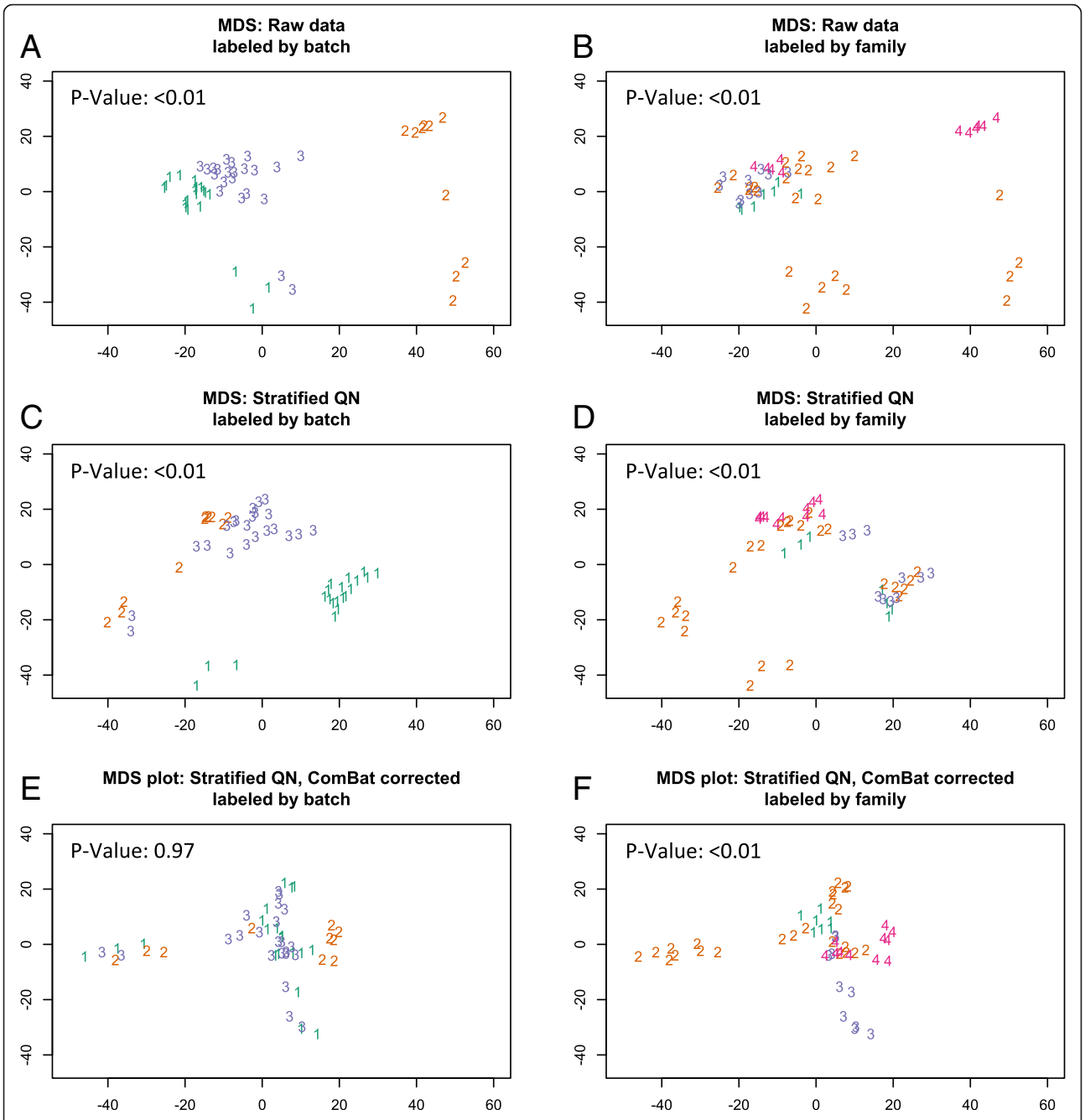


Fig. 2 Multidimensional scaling plots of *M* values by batch and family. Multidimensional scaling plots for raw (**a**, **b**), stratified QN (**c**, **d**) and ComBat stratified QN (**e**, **f**) *M* values. For each plot, the 1000 most variable probes were selected. In **a**, **c** and **e**, *numbers* represent batches and are *coloured* accordingly, with clustering by batch clearly seen in **a**, to a lesser extent in **c** and removed in **e**. In **b**, **d** and **f**, *numbers* represent family groups and are *coloured* accordingly with the clearest clustering present in **f** after the batch effect has been removed. *p* values are from an ANOVA test for significance of batch (**a**, **c** and **e**) or family (**b**, **d** and **f**) on the first principal component from a PCA on the beta values

technical replicates in addition to the three unique samples on each batch, to permit generation of data from analysis of the same biological sample. In data lacking technical bias, replicate samples would be expected to generate the most similar methylation profiles, while methylation profiles generated from closely related individuals should also cluster tightly compared to distantly or unrelated individuals. However, if technical bias such as a batch effect has been introduced, this distorts the profiles and samples no longer cluster by biological similarity but instead the most evident grouping would be by batch.

Batch effect (between array variation) was examined and the density distribution plot (Fig. 3a) of the raw β values from all three batches reveals significant bias. The greatest contributor to batch effect was the date on which the BeadChips were processed, with bisulphite conversion performed on the same day as BeadChip processing. Employing a stratified QN (Fig. 3b) and/or ComBat normalisation (Fig. 3c) dramatically reduced

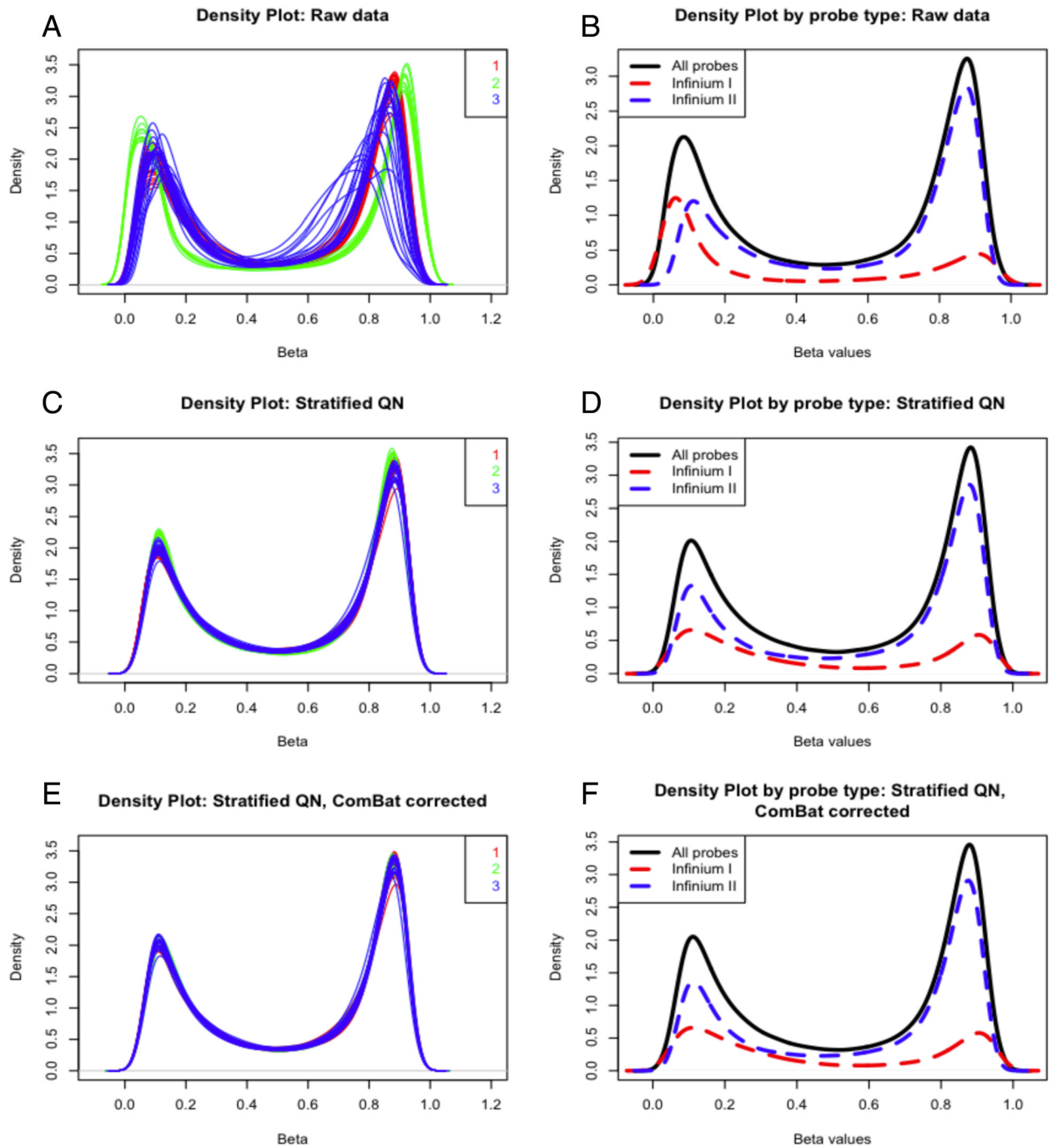


Fig. 3 Density distribution of β values. Density plot and probe distribution of β values for raw pre-normalisation data (**a**, **b**), after stratified QN (**c**, **d**) and with stratified QN and ComBat batch correction (**e**, **f**). For density plots (**a**, **c**, **e**), a *single line* represents a sample, with samples *coloured* by batch. A clear batch effect is present in **a**, lessened in **c** and removed in **e**. For the probe distribution (**b**, **d**, **f**), one sample has been chosen with the *red dashed line* indicating type I probe distribution, the *blue dashed line* type II and the *solid black line* the combined probe distribution. The probe type distribution is also improved after normalisation, as types I and II are more closely aligned in **d** and **f** compared to **b**

this observed effect. For between array biases, Fig. 3 shows the density distribution of β values for raw data samples (A), after stratified QN (C) and after stratified QN combined with ComBat correction (E). This is particularly evident when comparing the β value density plots of three groups of replicate samples (Additional file 2: Figure S1A, C and E).

Stratified QN also performs best at removing within array biases as the distribution of probe I and II types become more uniform (Fig. 3b, d, f). This bias is driven by the differing biochemistry of the probes, with type I employing a single colour channel with a different bead for methylated and unmethylated DNA and type II containing one bead in two colour channels. The underlying biology targeted by each probe is confounded by this technical bias, as type I measures CpG-dense regions (such as islands) while type II can only tolerate three CpGs in the length of the probe. As such, type I interrogates a greater proportion of unmethylated to methylated DNA, while type II performs the opposite. Removing the probe bias is imperative for accurate comparisons between these probe types when pooling probe I and II data, which is necessary for accurate genome-wide methylation information of both CpG rich and poor regions.

In contrast, the density plots of β values for other normalisation (SWAN and FunNorm) methods do not improve to the same degree and in some cases greater variation is introduced (Additional file 3: Figure S2C–G). For example, a worsening of the batch effect is seen for SWAN normalisation (Additional file 3: Figure S2D), compared to raw data (Additional file 2: Figure S1A) and the distribution of methylated and unmethylated signals is inverted following FunNorm (Additional file 3: Figure S2E).

The second approach employed to examine the performance of the normalisation methods was to generate multidimensional scaling (MDS) plots. These permitted the visualisation of the two-dimensional projection of the differences between samples. For each plot, the 1000 most variable probes were selected, as these represent the most pertinent biological differences between samples. *M* values were used as opposed to β values, the latter of which have been shown to suffer severe heteroskedasticity at very high and low values [18]. Again, a strong batch effect is observed in the raw data (Fig. 2a) as expected and this is removed or significantly reduced following normalisation using stratified QN (Fig. 2c) and ComBat (Fig. 2e) corrected data. The strong batch effect masks the familial relationships in the raw data; however, following the correction, clustering according to kinship is clearly evident. Similarly, the replicate samples (in Additional file 2: Figure S1), which group disparately in the raw data (A, B), co-locate or cluster tightly following stratified QN (C, D) and ComBat (E, F). The MDS plots for each normalisation method (Additional file 4: Figure S3) also show stratified QN followed by ComBat to be the most effective method for removing clustering by batch.

This efficacy of normalisation methods in reducing clustering of samples by batch was assessed quantitatively by ANOVA to test the effect of batch on the first principal component. The ANOVA was repeated for each normalisation method, using *M* values from the top 1000 most variable sites. Consistent with the visualised MDS plot, the *p* value was highly significant demonstrating the significant association of batch in *M* value in raw and stratified QN data ($p < 0.01$) but was not significant following correction using ComBat ($p = 0.97$).

For a final qualitative measure to examine effectiveness of between array normalisation, hierarchical cluster dendrograms were generated. Application of stratified QN and ComBat (Additional file 5: Figure S4) again demonstrated superior normalisation when visualised by this method; with raw data samples clearly clustering into three distinct groups (Additional file 5: Figure S4A), stratified QN resulting in improved clustering (B) while ComBat batch correction following stratified QN completely removes the batch effect (C) permitting the desired outcome with related individuals clustering together in familial groups. Furthermore, replicate samples cluster more clearly after ComBat normalisation (C, red stars) indicating removal of batch effects without perturbing biologically relevant information.

To quantitatively assess the performance of these normalisation methods, the median absolute difference in *M* values was calculated for six replicate pairs, with one sample from each pair interrogated on a separate batch. With the exception of one pair, stratified QN with ComBat was found to have the lowest absolute median difference between technical replicate pairs, corresponding to the highest correlation between replicate pairs (see Additional file 6: Table S2). While others such as SWAN introduced an increase in the error rate relative to the raw data values.

Finally, standard error measures for imprinted regions were calculated and compared between methods as described in the statistical analysis section of the methods. Smaller values indicate lower errors and more reliable data. A DMRSE of 0.0048 was calculated for the raw data, with this value increasing with following normalisations using QN (0.0052), noob (0.0052) and functional normalisation (0.0056). The remaining normalisation methods generated reduced DMSRE values with stratified QN with ComBat batch correction again producing the smallest error values at 0.0012. See Additional file 7: Table S3 for a full list of DMRSE values and Additional file 8: Figure S5 for the density plots of these probes.

Increased power for determining true biological associations

Critical to any normalisation method is the maintenance of true biological differences between samples. As described in the statistical analysis section of the methods, a previously identified meQTL was selected to perform association analysis with prior to and following normalisation. Following Bonferroni correction, a significant association was detected in the raw data (Fig. 4a, p value = 7.29e-06), increasing markedly after stratified QN (Fig. 4b, p value = 3.53e-07). After ComBat (C), there was a drop in significance compared to stratified QN and raw, yet the p value was still highly significant (p value = 1.05e-05) indicating preservation of the biological information of interest. The drop in significance after batch correction may be explained as confounding between batch and family, which is removed after ComBat. Ideally, samples would be randomised across experiments; however, the nature of familial studies is such that this is not always possible, as samples are collected at different time points, often across generations. To maintain maximum power, the inclusion of all available samples is essential and, therefore, data processing methods capable of dealing with non-ideal datasets are required.

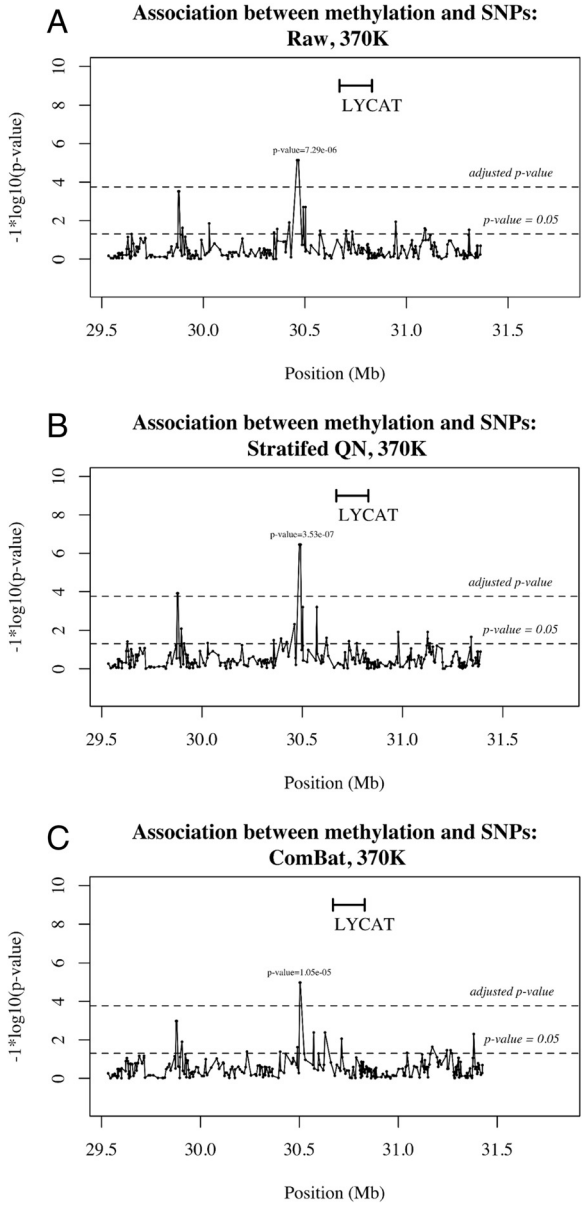


Fig. 4 Association plot between SNPs and methylation. Association between methylation at cg17749961 and SNPs in a 2-Mb window. There is a significant association in the raw data (**a**, p value = 7.29e-06) which increases after stratified QN (**b**, p value = 3.53e-07) and drops slightly after ComBat correction (**c**, p value = 1.05e-05)

Epigenome-wide methylation has long been shown to drift with age, specifically global hypomethylation and region-specific hypermethylation are observed [23]. The association between age and epigenome-wide methylation was compared for raw data, stratified QN normalised data and ComBat-corrected stratified QN data to demonstrate that this biological information was preserved after normalisation and batch correction. After normalisation (Additional file 9: Figure S6B), there are many more significant associations with age than in the raw data (Additional file 9: Figure S6A), indicated by a greater number of points above the expected line and a much greater Lambda value (median of observed –log10 p values divided by the median of expected –log10 p values), with an increase from 0.838 to 1.402. There is another small increase in significance after ComBat batch correction (Additional file 9: Figure S6C) to 1.448, again indicating improved strength in testing biological associations.

Discussion

There is currently a plethora of pre-processing methods and R packages available for analysis of 450k array data, and comprehensive review articles evaluating their utility have been published [24–26]. The majority of these are designed for specific types of sample sets, particularly those comprised of two distinct groups such as case–control or cancer-normal with substantial methylation differences between the two groups. For different datasets, such as those from familial studies, which include complex pedigree structures instead of two distinct groups, these methods may be ineffective or worse, detrimental in that they introduce technical bias, as identified with selected methods in this paper. To correctly normalise data, it is critical to choose the most appropriate method; yet there has been little focus on developing appropriate processing pipelines for familial methylation array analysis, despite the current interest in inherited drivers of methylation patterns. Further barriers are the various format requirements and the lack of integration to provide a seamless processing pipeline. Here, we have

tested eight different methods and presented a preliminary pre-processing pipeline for familial data (depicted in Fig. 5). This pipeline creates a template to guide and expedite the analysis of familial datasets, particularly generated using the 450k array data. Sample size ($n = 50$) is a limitation of this study, therefore additional familial studies would aid in validating the pipeline.

A fundamental requirement for processing methylation array data is effective adjustment for technical bias, including batch effects and adjusting for the two-probe biochemistry of the array. Batch effects may be introduced through bisulphite conversion or downstream processing or variation in array quality. Various methods have been developed to adjust for these effects, mostly involving variations in quantile normalisation, a technique commonly used in analysis of microarray datasets to align two different distributions so they result in identical statistical properties [26–29].

BMIQ and functional normalisation have been advocated as the preferred methods for cancer studies as they are more specific in design than quantile normalisation and have been shown to be more effective at removing unwanted technical bias [27, 29]. However, these methods work most effectively on case–control or tumour-normal datasets respectively and to the best of our knowledge, optimal pre-processing methods for

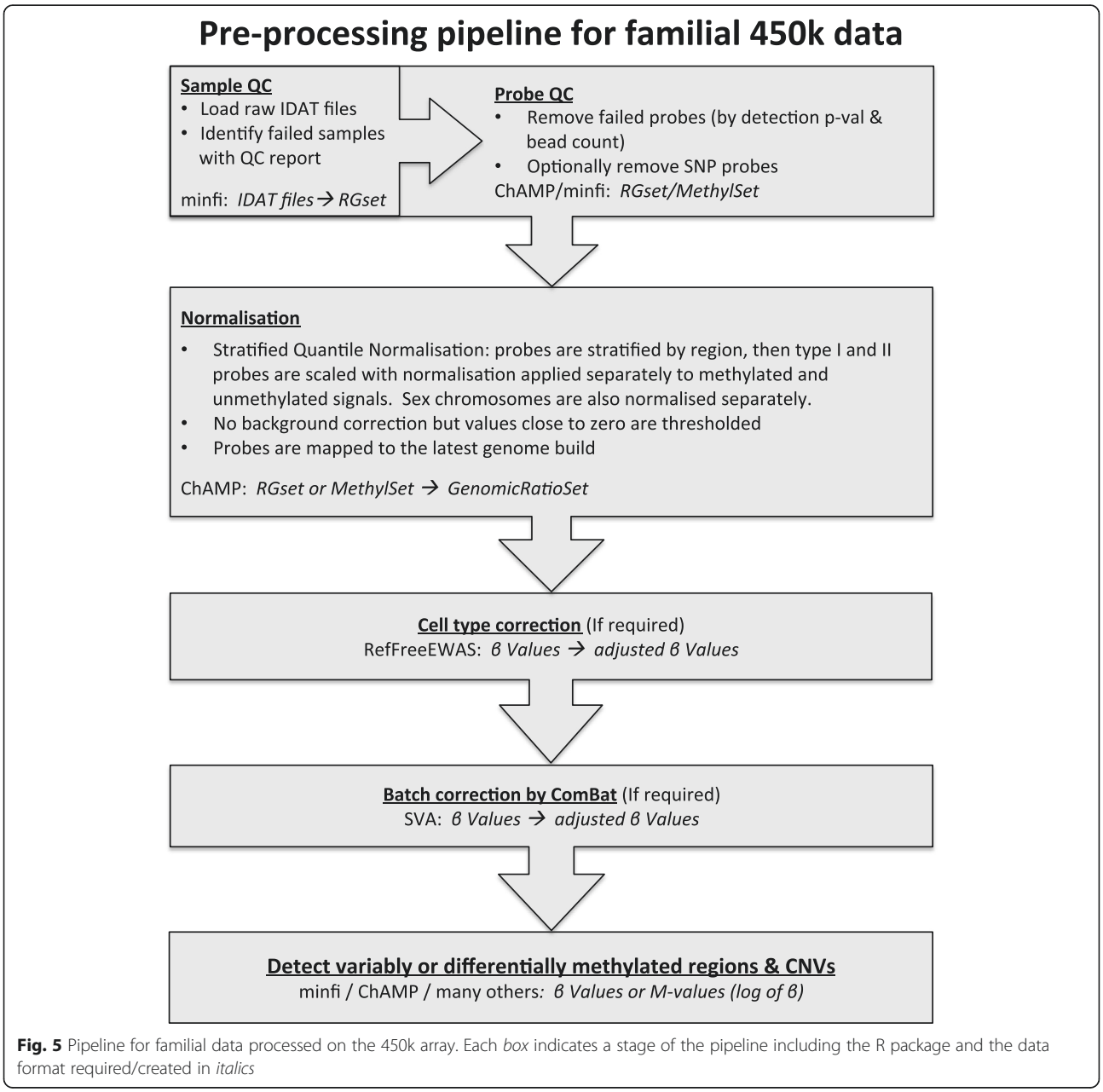


Fig. 5 Pipeline for familial data processed on the 450k array. Each *box* indicates a stage of the pipeline including the R package and the data format required/created in *italics*

familial-based data, such as performed here, have not been reported. Normalisation methods necessarily make assumptions about data, with the accuracy of these assumptions varying for different datasets. Thus, the same normalisation method can have a vastly different effect on different types of data and conversely, as shown here, different normalisation methods can have vastly different effects on the same data. It is therefore a key to select the right normalisation method for the dataset of interest. Of the eight methods tested, stratified QN was consistently identified as the best normalisation method across all visual and quantitative evaluation metrics for use in this context. The principle underpinning this normalisation is stratification by genomic region and is thus ideal for data where the differences between adjacent genomic loci are maintained. This is in contrast to tumour-normal tissue datasets where there are large blocks of dramatically altered methylation patterns throughout the tumour genome [30]. Again not surprisingly, packages that utilise differences in negative control methylation patterns between cases and controls such as FunNorm were not found to be effective on familial datasets where no 'normal' control is available.

The inherent strengths of familial data could be further exploited by a normalisation technique that accounts for known relationships between samples. Such a method could draw on pedigree information to ensure normalisation has effectively removed technical bias while maintaining known biologically relevant information such as relatedness and familial clustering by methylation. A diagnostic metric accounting for a known relationship could be used to test the efficacy of pre-processing methods in a similar manner to the standard error associated with iDMRs from the *wateRmelon* package.

It may also be of importance for researchers to consider the undesirable effect of non-specific binding and the presence of SNPs in the probe body. A study from the Weksberg lab found around 6 % of probes on the array cross-hybridised to non-targeted genomic regions [31]. They have catalogued these probes and suggest removing them prior to downstream analysis. Their study also demonstrates that SNPs in the probe body can interfere with probe binding, altering the methylation signal at around 14 % of sites. Illumina recommends all probes containing a SNP within 10 bp of the interrogated CpG site ought to be removed, while others suggest the 'probe effect' continues to the entire 50-bp length of the probe [31, 32]. The removal of all such probes would be undesirable for studies examining the effect of genotype on methylation, as evidence suggests the vast majority of these SNPs occur either at the CpG site itself (meSNPs) or close by [32].

To overcome this issue, Zhi and colleagues suggest an elegant approach to examine the effect of meSNPs on methylation without the potential bias introduced by SNPs altering probe binding [32]. The type II probes contain only one bead type for both methylated and unmethylated sites of interest, with the methylation status of the loci designated by the addition of a different coloured nucleotide (red or green) at the single base extension. As type II probes terminate one base pair before the cytosine of the CpG dinucleotide, a mutation at the cytosine itself would not affect probe binding. As such, probes without SNPs in the probe body but present at the single base extension can reliably be used to examine the effect of meSNPs on methylation, a very useful technique for examining the effect of inherited variation on methylation patterns.

Conclusions

Preservation of the biological integrity of information from methylation array data is imperative and requires appropriate pre-processing to minimise technical errors, which will be dictated by the type of data. Stratified QN in combination with ComBat batch correction performed the best of those methods tested for normalising familial data interrogated on 450k array. This method was observed to remove technical biases while maintaining biologically relevant information; allowing true biological differences and similarities to inform our search for the role of methylation patterns driving disease processes. The workflow presented in this paper (highlighted in Fig. 5) provides a streamlined methodology to pre-process familial data and may also be instructive for other datasets including longitudinal studies where the same individuals are repeatedly measured over time.

Additional files

Additional file 1: Table S1. Clinical data and sample extraction and storage information. (DOCX 20 kb)

Additional file 2: Figure S1. Hierarchical cluster dendrogram for raw, stratified QN and ComBat-corrected data. Samples are clustered by similarity and labelled by batch. Raw data samples (A) clearly cluster into three distinct batches while stratified QN (B) partially adjusts clustering by batch and stratified QN combined with ComBat considerably diminishes the batch effect (C). Red stars indicate replicate samples which cluster more clearly in (C), indicating removal of batch effects. (PDF 449 kb)

Additional file 3: Figure S2. Density distribution of β values and multidimensional scaling plots of *M* values for replicate samples. Density (A, C, E) and MDS (B, D, F) plots of three replicate sample groups for raw (A, B), stratified QN (C, D) and stratified QN ComBat-corrected (E, F) data. For all plots, samples are coloured by batch 1–3 as labelled. Density plots show the distribution of β values, which become more uniform after stratified QN (C) and stratified QN plus ComBat (E). MDS plots show clustering of the 1000 most variable sites by *M* value, highlighting the decreasing variance between replicate groups after stratified QN and ComBat (F). (PDF 7387 kb)

Additional file 4: Figure S3. Density distribution of β values for imprinted differentially methylated regions. Density plots for raw (A), stratified QN (C) and stratified QN with ComBat (E) for 227 probes mapping known imprinted differentially methylated regions. Each line represents a sample, with samples coloured by batch. As methylation at

these loci is allele-specific there is a single density distribution rather than the bimodal distribution seen in Additional file 3: Figure S2. The standard error-type measure (DMRSE) diminishes with Stratified QN and ComBat, indicating more reliable data. B, D and F show the Infinium I and II probe distributions, which becomes more uniform with stratified QN and ComBat. (PDF 4133 kb)

Additional file 5: Figure S4. Density distribution of β values for all normalisation methods. Density plots of β values for various normalisation methods: raw pre-normalisation data (A), quantile normalisation (B), BMIQ (C), SWAN (D), FunNorm (E), Dasen (F), noob (G), stratified QN (H), raw with ComBat correction (I) and stratified QN with ComBat correction (J). A single line represents a sample with samples coloured by batch. The batch effect present in the raw data (A) remains after the majority of normalisation methods with Dasen (F) and stratified QN (H) showing the most uniform distributions. Some methods such as quantile normalisation (B) and FunNorm (E) flip the methylated and unmethylated signal distribution. ComBat is effective at removing batch effects in both raw (I) and normalised (J) data, with the best outcome seen with stratified QN with ComBat batch correction (J). (PDF 260 kb)

Additional file 6: Table S2. Median absolute difference between technical replicate pairs. (DOCX 14 kb)

Additional file 7: Table S3. Standard error measures for imprinted differentially methylated regions for the various normalisation methods. (DOCX 13 kb)

Additional file 8: Figure S5. Multidimensional scaling plots of *M* values by batch for all normalisation methods. Multidimensional scaling plots for raw (A), quantile normalisation (B), BMIQ (C), SWAN (D), FunNorm (E), Dasen (F), noob (G), stratified QN (H), raw with ComBat correction (I) and stratified QN with ComBat correction (J). For each plot, the 1000 most variable probes were selected. Batches are numbered and coloured, with clustering by batch clearly seen in the raw data (A) and removed to varying degrees with different normalisation methods. ComBat correction following stratified QN provides optimal batch correction removal as the samples no longer cluster according to batch. (PDF 559 kb)

Additional file 9: Figure S6. QQ plots for the association of age and epigenome-wide methylation. QQ plots with −log10 *p* values from the linear model of methylation and age plotted against expected −log10 *p* values. Raw data (A), data normalised by stratified QN (B) and data normalised by stratified QN then corrected with ComBat (C). (PDF 85 kb)

Abbreviations
CpG, cytosine-guanine pair; meQTL, methylation quantitative trait loci; MDS, multidimensional scaling; meSNPs, methylation single nucleotide polymorphisms

Acknowledgements
The authors would like to thank Dr Alicia Oschlack for her helpful discussions at the commencement of this project. The authors wish to thank the participants of the Tasmanian Familial Prostate Cancer Study. In addition, we would like to extend our thanks to the Royal Hobart Hospital Cancer Auxillary members for their support of EC over the course of her studies.

Funding
Support for this work was provided by Cancer Australia, Cancer Council Tasmania and the Royal Hobart Hospital Cancer Auxillary. JLD is supported by an Australian Research Council Future Fellowship. Funding bodies have had no input into the design, analysis or preparation of this manuscript.

Authors' contributions
EC conducted the primary research, performed the laboratory analyses, and drafted the manuscript. RT participated in the study design and provided direction for the statistical analysis. JM provided the molecular laboratory support. AH participated in the study design and aided in drafting the manuscript. JC participated in the study design, provided assistance with the analysis and aided in drafting the manuscript. JLD participated in the study design and was substantially involved in drafting the manuscript. All authors read and approved the final manuscript.

Competing interests
The authors declare that they have no competing interests.

Consent for publication
Informed consent included the consent to publish an individual's de-identified data.

Ethics approval and consent to participate
Informed consent was obtained from all participants, following ethics approval from the Human Research Ethics Committee Tasmania Network (H009999).

Author details
[1]Menzies Institute for Medical Research, University of Tasmania, Private Bag 23 Medical Sciences Building 2, Hobart, TAS, Australia. [2]Centre for Research in Mathematics, School of Computing, Engineering and Mathematics, Western Sydney University, Parramatta Campus, Locked Bag 1797, Penrith, NSW 2751, Australia. [3]School of Medicine, University of Tasmania, Medical Sciences Building 2, Hobart, TAS 7001, Australia.

Received: 3 April 2016 Accepted: 26 June 2016
Published online: 16 July 2016

References
1. Suzuki MM, Bird A. DNA methylation landscapes: provocative insights from epigenomics. Nat Rev Genet. 2008;9:465–76.
2. Bock C. Analysing and interpreting DNA methylation data. Nature Reviews Genetics13, 705-719 (October 2012) | doi:10.1038/nrg3273.
3. Ji H, Ehrlich LIR, Seita J, Murakami P, Doi A, Lindau P, et al. Comprehensive methylome map of lineage commitment from haematopoietic progenitors. Nature. 2010;467:338–42.
4. Irizarry RA, Ladd-Acosta C, Wen B, Wu Z, Montano C, Onyango P, et al. The human colon cancer methylome shows similar hypo- and hypermethylation at conserved tissue-specific CpG island shores. Nat Genet. 2009;41:178–86.
5. Gertz J, Varley KE, Reddy TE, Bowling KM, Pauli F, Parker SL, et al. Analysis of DNA methylation in a three-generation family reveals widespread genetic influence on epigenetic regulation. Bickmore WA, editor. PLoS Genet. 2011;7, e1002228.
6. Drong AW, Nicholson G, Hedman AK, Meduri E, Grundberg E, Small KS, et al. The presence of methylation quantitative trait loci indicates a direct genetic influence on the level of DNA methylation in adipose tissue. PLoS One. 2013;8, e55923.
7. Bell JT, Pai AA, Pickrell JK, Gaffney DJ, Pique-Regi R, Degner JF, et al. DNA methylation patterns associate with genetic and gene expression variation in HapMap cell lines. Genome Biol. 2011;12:R10.
8. Gibbs JR, van der Brug MP, Hernandez DG, Traynor BJ, Nalls MA, Lai S-L, et al. Abundant quantitative trait loci exist for DNA methylation and gene expression in human brain. Flint J, editor. PLoS Genet Public Library of Science. 2010;6:e1000952.
9. Smith AK, Kilaru V, Kocak M, Almli LM, Mercer KB, Ressler KJ, et al. Methylation quantitative trait loci (meQTLs) are consistently detected across ancestry, developmental stage, and tissue type. BMC Genomics BioMed Central Ltd. 2014;15:145.
10. Zhang D, Cheng L, Badner JA, Chen C, Chen Q, Luo W, et al. Genetic control of individual differences in gene-specific methylation in human brain. The American Journal of Human Genetics The American Society of Human Genetics. 2010;86:411–9.
11. Shen H, Fridley BL, Song H, Lawrenson K, Cunningham JM, Ramus SJ, et al. Epigenetic analysis leads to identification of HNF1B as a subtype-specific susceptibility gene for ovarian cancer. Nat Commun. 2013;4:1628. doi: 10.1038/ncomms2629.
12. Ward RL, Dobbins T, Lindor NM, Rapkins RW, Hitchins MP. Identification of constitutional MLH1 epimutations and promoter variants in colorectal cancer patients from the Colon Cancer Family Registry. Genet Med. 2013;15:25–35.
13. Sharma S, Kelly TK, Jones PA. Epigenetics in cancer. Carcinogenesis Oxford University Press. 2010;31:27–36.
14. R Core Team. R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. 2014. http://www.R-project.org/
15. Aryee MJ, Jaffe AE, Corrada-Bravo H, Ladd-Acosta C, Feinberg AP, Hansen KD, et al. Minfi: a flexible and comprehensive bioconductor package for the analysis of Infinium DNA methylation microarrays. Bioinformatics Oxford University Press. 2014;30:1363–9.
16. Davis S, Du P, Bilke S, Triche T, Bootwalla M. Methylumi: handle illumina methylation data. 2012: R package version 2.12.0

17. Morris TJ, Butcher LM, Feber A, Teschendorff AE, Chakravarthy AR, Wojdacz TK, et al. ChAMP: 450 k chip analysis methylation pipeline. Bioinformatics Oxford University Press. 2014;30:428–30.
18. Du P, Zhang X, Huang C-C, Jafari N, Kibbe W, Hou L, et al. Comparison of Beta-value and M-value methods for quantifying methylation levels by microarray analysis. BMC Bioinformatics BioMed Central Ltd. 2010;11:587.
19. Leek JT, Johnson WE, Parker HS, Jaffe AE, Storey JD. The sva package for removing batch effects and other unwanted variation in high-throughput experiments. Bioinformatics Oxford University Press. 2012;28:882–3.
20. Pidsley R, Y Wong CC, Volta M, Lunnon K, Mill J, Schalkwyk LC. A data-driven approach to preprocessing Illumina 450 K methylation array data. BMC Genomics. 2013;14:293.
21. Shoemaker R, Deng J, Wang W, Zhang K. Allele-specific methylation is prevalent and is contributed by CpG-SNPs in the human genome. Genome Research Cold Spring Harbor Lab. 2010;20:883–9.
22. GenABEL project developers. GenABEL: genome-wide SNP association analysis. R package version 1.8-0. 2013.
23. Jung M, Pfeifer GP. Aging and DNA methylation. BMC Biology 2015 13:1. BioMed Central; 2015;13:1
24. Touleimat N, Tost J. Complete pipeline for Infinium(®) Human Methylation 450 K BeadChip data processing using subset quantile normalization for accurate DNA methylation estimation. Epigenomics. 2012;4:325–41.
25. Marabita F, Almgren M, Lindholm ME, Ruhrmann S, Fagerström-Billai F, Jagodic M, et al. An evaluation of analysis pipelines for DNA methylation profiling using the Illumina HumanMethylation450 BeadChip platform. Epigenetics Landes Bioscience. 2013;8:333–46.
26. Morris TJ, Beck S. Analysis pipelines and packages for Infinium HumanMethylation450 BeadChip (450 k) data. Methods Elsevier Inc. 2015;72:3–8.
27. Teschendorff AE, Marabita F, Lechner M, Bartlett T, Tegnér J, Gomez-Cabrero D, et al. A beta-mixture quantile normalization method for correcting probe design bias in Illumina Infinium 450 k DNA methylation data. Bioinformatics Oxford University Press. 2013;29:189–96.
28. Sun Z, Chai HS, Wu Y, White WM, Donkena KV, Klein CJ, et al. Batch effect correction for genome-wide methylation data with Illumina Infinium platform. BMC Med Genomics BioMed Central Ltd. 2011;4:84.
29. Fortin JP, Labbe A, Lemire M, Zanke BW, Hudson TJ. Functional normalization of 450 k methylation array data improves replication in large cancer studies. Genome Biology, February 2014 doi: 10.1186/s13059-014-0503-2.
30. Timp W, Bravo HC, McDonald OG, Goggins M, Umbricht C, Zeiger M, et al. Large hypomethylated blocks as a universal defining epigenetic alteration in human solid tumors. Genome Med. 2014;6
31. Chen Y-A, Lemire M, Choufani S, Butcher DT, Grafodatskaya D, Zanke BW, et al. Discovery of cross-reactive probes and polymorphic CpGs in the Illumina Infinium HumanMethylation450 microarray. Epigenetics. 2013;8:203–9.
32. Zhi D, Aslibekyan S, Irvin MR, Claas SA, Borecki IB, Ordovás JM, et al. SNPs located at CpG sites modulate genome-epigenome interaction. Epigenetics. 2013;8(8):802-6.
33. Du P, Kibbe WA, Lin SM. lumi: a pipeline for processing Illumina microarray. Bioinformatics Oxford University Press. 2008;24:1547–8.
34. Maksimovic J, Gordon L, Oshlack A. SWAN: Subset-quantile within array normalization for Illumina Infinium HumanMethylation450 BeadChips. Genome Biol. 2012;13:R44.
35. Triche TJ, Weisenberger DJ, Van Den Berg D, Laird PW, Siegmund KD. Low-level processing of Illumina Infinium DNA Methylation BeadArrays. Nucleic Acids Res Oxford University Press. 2013;41:e90.
36. Gagnon-Bartsch JA, Jacob L, Speed TP. Removing unwanted variation from high dimensional data with negative controls. 2012. Technical Report, UC Berkeley. Technical report 820, p. 1–104.

Appendix 4.1 **Linkage regions with the highest LOD scores previously identified through the** Tasmanian Familial Prostate Cancer Study

SNP	Chr	Position	LOD Score
RS2600776	2	237263622	3.09
RS2600793	2	237269550	3.25
RS7585432	2	237291473	3.44
RS975919	2	237293119	3.44
RS10929183	2	237293840	3.61
RS7556982	2	237308938	3.67
RS13419340	2	237325521	3.67
RS1549869	2	237335771	3.7
RS999031	2	237355157	3.58
RS1897464	2	237363415	3.61
RS934397	2	237371678	3.61
RS1344762	2	237373226	3.73
RS10929194	2	237385373	3.73
RS13000320	2	237388433	3.73
RS1368202	2	237395901	3.85
RS10166334	2	237405247	3.91
RS1000007	2	237416793	3.88
RS7565608	2	237425567	4.03
RS11694315	2	237435644	4.03
RS3888913	2	237444863	3.41
RS30105	2	237459284	3.43
RS6737351	2	237464699	3.52
RS10169105	2	237470609	3.37

SNP	Chr	Position	LOD Score
RS4714457	6	41485970	2.26
RS814836	6	41497306	2.2
RS814839	6	41502190	2.2
RS13219238	6	41506909	2.16
RS13196762	6	41512003	2.22
RS766389	6	41514764	2.08
RS11962670	6	41520542	2.04
RS9381065	6	41522528	2.02
RS912881	6	41523715	2.02
RS1970923	6	41525198	1.98
RS9357362	6	41541193	2.02
RS2496635	6	41542795	2.11
RS7762551	6	41545237	2.35
RS728825	6	41551451	2.43
RS728826	6	41551593	2.58
RS2477831	6	41551667	2.58
RS10947972	6	41554327	2.58
RS4714468	6	41560974	2.58
RS2496637	6	41563034	2.35
RS721313	6	41563275	2.31
RS3800282	6	41568265	2.35
RS6913778	6	41572879	2.44
RS6907558	6	41575838	2.59

SNP	Chr	Position	LOD Score
RS1722791	15	21503831	2.17
RS1524842	15	21505342	2.43
RS1717839	15	21525066	2.36
RS11632341	15	21540364	2.36
RS1628195	15	21551731	2.41
RS824205	15	21559164	2.27
RS8182040	15	21569096	2.25
RS1722842	15	21582049	2.38
RS12439817	15	21582093	2.41
RS7181322	15	21583164	3.16
RS940595	15	21589360	3.13
RS8038234	15	21597810	3.13
RS824211	15	21598281	3.16
RS7180295	15	21601351	3.01
RS2352765	15	21606718	3.01
RS844033	15	21609833	3.01
RS824162	15	21609981	2.99
RS1524845	15	21612590	3.04
RS824167	15	21614609	3.01
RS2177077	15	21618739	3.01
RS8031166	15	21633826	2.68
RS11633486	15	21639815	2.68
RS8038712	15	21647999	2.65

RS12463648	2	237475305	3.37
RS4663682	2	237481662	3.37
RS7603272	2	237486328	3.48
RS7581699	2	237496143	3.48
RS10929201	2	237497105	3.6
RS729454	2	237516077	3.65
RS1435847	2	237527888	3.56
RS7600637	2	237538697	3.45
RS2701323	2	237550613	3.28
RS4663691	2	237565291	3.25
RS2701336	2	237571642	3.2
RS4663692	2	237573371	3.14
RS755512	2	237575821	3.12
RS2573718	2	237582343	3.2
RS2318131	2	237598705	3.14
RS7591958	2	237608324	3.12
RS7576705	2	237612027	3.14
RS7597414	2	237622493	3.26
RS7589198	2	237632354	3.21
RS12620999	2	237701106	3.18
RS10172321	2	237730202	3.35
RS6714237	2	237734087	3.32
RS7599969	2	237734970	3.32
RS12613316	2	237740231	3.18
RS4527163	2	237765225	3.15
RS4663242	2	237767644	3.15
RS4233629	2	237768857	3.08

RS2495232	6	41582705	2.77
RS2495233	6	41583425	2.77
RS2477842	6	41597419	2.85
RS2496652	6	41612104	2.95
RS4714484	6	41639201	2.95
RS913074	6	41646523	3.1
RS4714487	6	41655290	3.07
RS13362583	6	41671135	3.07
RS913075	6	41676884	2.91
RS9381084	6	41678328	2.75
RS9369298	6	41684618	2.75
RS2842639	6	41690967	2.75
RS6928533	6	41701974	2.8
RS1973920	6	41711172	2.77
RS2495229	6	41713808	2.73
RS2268408	6	41718697	2.77
RS2842658	6	41727028	2.89
RS4714501	6	41727341	2.94
RS2230088	6	41729249	2.97
RS2143678	6	41731011	2.67
RS1474762	6	41735030	2.56
RS1474761	6	41735041	2.68
RS4714503	6	41741250	2.63
RS6458234	6	41746737	2.46
RS1011101	6	41751364	2.12
RS1883816	6	41752419	2.14

RS1459958	15	21649687	2.45
RS10519445	15	21657901	2.47
RS4778341	15	21663374	2.47
RS1459985	15	21668350	2.47
RS2883186	15	21680981	2.21
RS4778346	15	21686776	2.11
SNP	**Chr**	**Position**	**LOD Score**
RS738092	22	19190931	3.03
RS7291930	22	19206509	4.64
RS1110462	22	19211385	4.6
RS5995708	22	19217080	4.47
RS7292126	22	19226926	4.47
RS886319	22	19233410	4.51
RS177421	22	19243757	4.51
RS165674	22	19258809	4.6
RS361646	22	19277274	4.8
RS165626	22	19284760	4.7
RS552823	22	19286906	3.87
RS561595	22	19290707	3.9
RS680548	22	19295555	3.87
RS473304	22	19297450	3.64
RS654526	22	19303386	3.64

Appendix 4.2 **Significant me-QTL Associations for the Variable Methylation Approach using**

	CpG Name	Significant Associations	Highest -log10(p-value)
1	cg13387643	10	33.80
2	cg20592836	44	30.25
3	cg25203245	37	28.94
4	cg13928473	18	28.86
5	cg15083522	42	28.34
6	cg03075889	27	27.84
7	cg07414487	66	27.73
8	cg05792312	10	27.68
9	cg09281805	10	27.00
10	cg08146865	42	26.84
11	cg10724632	20	26.55
12	cg09084244	31	26.38
13	cg21927991	20	26.25
14	cg25013753	15	26.25
15	cg08210706	24	25.90
16	cg16490124	29	25.56
17	cg19393008	23	25.53
18	cg20205188	12	25.39
19	cg05809586	22	25.11
20	cg17723206	11	24.96
21	cg23098789	6	24.48
22	cg04145681	46	24.42
23	cg27481428	23	24.24
24	cg23681001	17	24.23

	CpG Name	Significant Associations	Highest -log10(p-value)
51	cg19300401	18	20.91
52	cg05161773	18	20.68
53	cg23052585	29	20.67
54	cg18618432	28	20.59
55	cg11251367	9	20.42
56	cg25543264	36	20.29
57	cg11585022	34	20.28
58	cg18527716	25	19.70
59	cg02113055	33	19.63
60	cg18572898	138	19.57
61	cg09993319	9	19.40
62	cg18709904	29	19.26
63	cg18088486	29	19.05
64	cg04610028	14	18.94
65	cg22851875	12	18.85
66	cg05338731	33	18.81
67	cg12551908	20	18.80
68	cg26128129	8	18.64
69	cg07686394	29	18.61
70	cg04028540	9	18.48
71	cg05059349	24	17.93
72	cg27341708	29	17.10
73	cg05509228	9	16.97
74	cg12186981	36	16.95

25	cg07240846	7	24.10
26	cg06330797	14	23.95
27	cg00231519	27	23.87
28	cg26705599	11	23.81
29	cg19360212	27	23.77
30	cg00257789	32	23.74
31	cg18828306	24	23.73
32	cg18624102	29	23.64
33	cg25674027	32	23.39
34	cg04998327	13	23.33
35	cg10530344	20	22.90
36	cg06318935	25	22.73
37	cg05134736	12	22.59
38	cg22274273	9	22.58
39	cg01127608	4	22.29
40	cg15567368	1	22.03
41	cg02978201	19	22.00
42	cg15765638	12	21.88
43	cg01341801	31	21.53
44	cg01891583	12	21.53
45	cg24009806	8	21.48
46	cg16791832	13	21.35
47	cg09856996	5	21.34
48	cg12342501	15	21.30
49	cg02890259	23	21.18
50	cg12657416	13	21.02

75	cg25593194	37	16.73
76	cg09533869	14	16.57
77	cg06032337	59	16.29
78	cg02658043	18	16.26
79	cg21498547	8	15.76
80	cg17056069	12	15.69
81	cg02533724	8	15.52
82	cg00345083	12	15.03
83	cg16748433	26	14.23
84	cg20536971	27	14.03
85	cg26365090	6	12.53
86	cg12195446	19	12.48
87	cg17662493	31	12.33
88	cg08238375	45	12.04
89	cg14797147	15	11.59
90	cg13232075	20	11.50
91	cg10507965	13	10.84
92	cg07501029	7	10.56
93	cg20086657	28	10.09
94	cg00704664	9	9.69
95	cg03796003	11	9.43
96	cg24925741	9	8.04
97	cg03224005	14	7.98
98	cg26642774	8	7.76
99	cg25465065	26	6.85
100	cg09289202	11	6.21

Mean Number of Significant hits: 21.94

Mean log p-value: 22.06

Appendix 4.3 **Significant me-QTL Associations for the Variable Methylation Approach using 95%-Reference Range**

	CpG Name	Significant Associations	Highest -log10(p-value)
1	cg20592836	44	30.25
2	cg25203245	37	28.94
3	cg15083522	42	28.34
4	cg07414487	66	27.73
5	cg02464073	9	27.35
6	cg08146865	42	26.84
7	cg10724632	20	26.55
8	cg23698271	7	26.33
9	cg21927991	20	26.25
10	cg06464078	22	25.81
11	cg16490124	29	25.56
12	cg19393008	23	25.53
13	cg17723206	11	24.96
14	cg24801230	57	24.70
15	cg23098789	6	24.48
16	cg04145681	46	24.42
17	cg27481428	23	24.24
18	cg23681001	17	24.23
19	cg07240846	7	24.10
20	cg06330797	14	23.95
21	cg26705599	11	23.81
22	cg19360212	27	23.77
23	cg00257789	32	23.74
24	cg18828306	24	23.73

	CpG Name	Significant Associations	Highest -log10(p-value)
49	cg13885788	7	19.76
50	cg18527716	25	19.70
51	cg02113055	33	19.63
52	cg18572898	138	19.57
53	cg08049519	35	19.55
54	cg09993319	9	19.40
55	cg00474373	7	19.39
56	cg18709904	29	19.26
57	cg18088486	29	19.05
58	cg04610028	14	18.94
59	cg05338731	33	18.81
60	cg07686394	29	18.61
61	cg05059349	24	17.93
62	cg27341708	29	17.10
63	cg07498088	42	16.99
64	cg12186981	36	16.95
65	cg04131969	19	16.73
66	cg09533869	14	16.57
67	cg06032337	59	16.29
68	cg02658043	18	16.26
69	cg24534774	35	16.22
70	cg21498547	8	15.76
71	cg17056069	12	15.69
72	cg04627110	59	15.61

25	cg25674027	32	23.39
26	cg04998327	13	23.33
27	cg10530344	20	22.90
28	cg06318935	25	22.73
29	cg22274273	9	22.58
30	cg01127608	4	22.29
31	cg04003990	15	22.03
32	cg15567368	1	22.03
33	cg16963093	3	22.00
34	cg21463262	7	21.92
35	cg05331763	10	21.83
36	cg01891583	12	21.53
37	cg16791832	13	21.35
38	cg09856996	5	21.34
39	cg12342501	15	21.30
40	cg02890259	23	21.18
41	cg12657416	13	21.02
42	cg19300401	18	20.91
43	cg05161773	18	20.68
44	cg23052585	29	20.67
45	cg18618432	28	20.59
46	cg05111645	115	20.46
47	cg11251367	9	20.42
48	cg10528424	4	20.26

73	cg02533724	8	15.52
74	cg10140678	12	14.62
75	cg25755428	6	14.43
76	cg16748433	26	14.23
77	cg04657146	1	13.44
78	cg23603995	26	13.25
79	cg03075889	21	13.16
80	cg26365090	6	12.53
81	cg12195446	19	12.48
82	cg17662493	31	12.33
83	cg14797147	15	11.59
84	cg07501029	7	10.56
85	cg20086657	28	10.09
86	cg00704664	9	9.69
87	cg03796003	11	9.43
88	cg27126508	13	9.03
89	cg24925741	9	8.04
90	cg03224005	14	7.98
91	cg11607219	22	7.75
92	cg25465065	26	6.85
93	cg02100397	2	6.73
94	cg09289202	11	6.21
95	cg21550016	30	5.86
96	cg01201512	13	5.64

Mean Number of Significant hits: 22.77

Mean log p-value: 18.9

Appendix 4.4 **The most significant associations from the risk loci approacl**

A) Risk loci identified through the Tasmanian Familial Prostate Cancer Stud

	Chr	Genomic Position	CpG Name
1	chr15	24125985	cg12151888
2	chr6	41407766	cg26223899
3	chr15	24043142	cg26261358
4	chr6	41383225	cg10863737
5	chr6	41528198	cg03036702
6	chr2	237992612	cg16995742

B) Risk Loci identified through published familial prostate cancer studies

	Chr	Genomic Position	CpG Name
1	chr1	235292369	cg16490124
2	chr1	241800323	cg03964373
3	chr1	240620177	cg11251367
4	chr1	233089275	cg00069771
5	chr1	233518998	cg16675926
6	chr1	232086152	cg23209941
7	chr1	231820076	cg07134368
8	chr1	242002464	cg24361198

C) Risk Loci identified through published prostate cancer GWAS

	Chr	Genomic Position	CpG Name
1	chr8	128079561	cg11123619
2	chr19	51362954	cg04741880
3	chr2	121684535	cg26075039
4	chr10	101910498	cg20720056
5	chr1	154839813	cg06221963
6	chr1	154839909	cg09359103
7	chr6	6543402	cg23069046
8	chr6	153455993	cg02956194
9	chr6	41528198	cg03036702
10	chr5	1298644	cg12474444
11	chr8	143751801	cg24634471
12	chr8	143757498	cg04035553
13	chr8	143751796	cg10596483
14	chr19	51336166	cg14773235
15	chr12	47353065	cg18468917
16	chr1	117487269	cg16060930

17	chr6	32186049	cg00366603
18	chr6	32188822	cg10158182
19	chr6	32202844	cg17239008
20	chr6	32186244	cg17351927
21	chr6	32415210	cg06281714
22	chr5	1325588	cg06550200
23	chr10	100167465	cg26690318
24	chr10	126700684	cg14375985
25	chr10	126686762	cg09349613
26	chr20	62387416	cg13301327
27	chr20	61660810	cg08564027
28	chr11	2211939	cg19586845
29	chr11	2243973	cg01452169
30	chr11	2222912	cg07146321
31	chr11	2212225	cg08241307
32	chr1	38180356	cg06437931
33	chr2	238410067	cg14271023
34	chr2	238380390	cg14458575
35	chr2	238392110	cg16989719
36	chr3	113254986	cg13284789

Variable Methylation Approach to Identify meQTLs

1. Identify most variable CpGs
2. Draw SNPs in 250kb surrounds
3. Perform association analysis in windows
4. Sort and organise most significant associations

```
############################################################
```

```
# Methods of determining the most variable CpG sites between
individuals
load("/Users/ecazaly/Desktop/PhD_Analysis/Association_2015april/
Ass_meth_input/R_workspaces_scripts_Meth/Methylation.RData") # load
clean data
# use the object Meth_M which contains normalised and batch
corrected M values (logit of Beta values) for 47 samples at 467263
CpG sites.  M values are used to determine which CpGs are the most
variable as this measure is less vulnerable to heteroskedasticity
than  Beta values. Once CpG sites are identified, Beta values can
then be used for more biologically interpretable plots


## Initial Comparison of variability methods ##
# check the difference between using M and Beta values in the SD
approach
length(which(rownames(Top100_SD_M) %in% rownames(Top100_SD))) # 71
cross over
# check which aren't in overlap and see what plots look like, some
look messy
non_overlap <- as.matrix(Top100_SD[which(!(rownames(Top100_SD_M) %in
% rownames(Top100_SD))),])
length(non_overlap) #29
rownames(non_overlap)
## Top 100/500 variable by 95% reference range, only sbe SNPs, no
probe SNPs
quantile_95_M_sbeSNP <- apply(Meth_M_sbeSNP,1, quantile,
probs=c(0, .025, 0.5, 0.975, 1))
range_95_M_sbeSNP <- as.matrix(quantile_95_M_sbeSNP[4,]-
quantile_95_M_sbeSNP[2,])
ord_range_95_M_sbeSNP <- range_95_M_sbeSNP[order(range_95_M_sbeSNP,
decreasing=TRUE),]
Top100_range_M_sbeSNP <- as.matrix(ord_range_95_M_sbeSNP[1:100])
Top500_range_M_sbeSNP <- as.matrix(ord_range_95_M_sbeSNP[1:500])
# check the difference between different methods
length(which(rownames(Top100_SD) %in% rownames(Top100_range_M))) #
51, 51%
length(which(rownames(Top100_SD_M) %in% rownames(Top100_range_M))) #
67, 67%
length(which(rownames(Top500_SD_M) %in% rownames(Top500_range_M))) #
407, 81%
```

```
length(which(rownames(Top100_SD_Meth_M_sbeSNP) %in%
rownames(Top100_range_M_sbeSNP))) # 73, 73%
length(which(rownames(Top500_SD_Meth_M_sbeSNP) %in%
rownames(Top500_range_M_sbeSNP))) # 428, 86%
# check the difference between CpGs with a SNP at the sbe and the
ones without taking this into account
length(which(rownames(Top100_SD_M) %in%
rownames(Top100_SD_Meth_M_sbeSNP))) # 80, 80%
length(which(rownames(Top500_SD_M) %in%
rownames(Top500_SD_Meth_M_sbeSNP))) # 396, 79%
length(which(rownames(Top100_range_M) %in%
rownames(Top100_range_M_sbeSNP))) # 90, 90%  # these have the
greatest overlap
length(which(rownames(Top500_range_M) %in%
rownames(Top500_range_M_sbeSNP))) # 415, 83%
# Thus, the range is better at pulling out the sbe SNPs rather than
SD, especially in the most variable sites (Top100) where its 10%
more than the Top500
# write a csv with 4 columns listing 100s/500s and the overlap
between
Tops_100 <- cbind(rownames(Top100_SD_M), rownames(Top100_range_M),
rownames(Top100_SD_Meth_M_sbeSNP), rownames(Top100_range_M_sbeSNP))
colnames(Tops_100) <- c("Top100_SD_M", "Top100_range_M",
"Top100_SD_Meth_M_sbeSNP", "Top100_range_M_sbeSNP")
write.csv(Tops_100, file="/Users/ecazaly/Desktop/PhD_Analysis/
Association_2015april/Ass_meth_input/Tops_100.csv")
Tops_500 <- cbind(rownames(Top500_SD_M), rownames(Top500_range_M),
rownames(Top500_SD_Meth_M_sbeSNP), rownames(Top500_range_M_sbeSNP))
colnames(Tops_500) <- c("Top500_SD_M", "Top500_range_M",
"Top500_SD_Meth_M_sbeSNP", "Top500_range_M_sbeSNP")
write.csv(Tops_500, file="/Users/ecazaly/Desktop/PhD_Analysis/
Association_2015april/Ass_meth_input/Tops_500.csv")
### Chose M over Beta values due to heteroskedasticity, chose a
combination of SD and 95%-range to give wider coverage and chose a
combination of only sbe SNP CpGs (SD) and CpGs unfiltered for sbe
SNPs (95%-range) ###



############################
## 1. Standard Deviation  ##
############################

# Prioritise the CpGs that are TypeII and have a SNP at the sbe as
methylation at these sites are the most likely to be affected by
SNPs but those SNPs will not affect the pribe binding.  Also check
for SNPs in probe body as these may affect the binding of the probe

load("/Users/ecazaly/Desktop/PhD_Analysis/Association_2015april/
Ass_meth_input/R_workspaces_scripts_Meth/Methylation.RData")
# Which are type II probes?
typeII <- Annotated_meth[which(Annotated_meth$Type=="II"),]
dim(typeII)  #336539     33
```

```
# Which of these have a SNP at the sbe?
typeII_sbeSNP <- typeII[which(!(typeII$SBE_rs=="NA")),]
dim(typeII_sbeSNP)  #7049   33
#What are the CpG names for these sites?
names_typeII_sbeSNP <- rownames(typeII_sbeSNP)
length(names_typeII_sbeSNP)  #7049
# Which ones also have a snp in the probe body?
typeII_sbeSNP_probeSNP <- typeII_sbeSNP[which(!(typeII_sbeSNP
$Probe_rs=="NA")),]
dim(typeII_sbeSNP_probeSNP) # none

# Which are the most variable between samples by Standard Deviation?
# First, get the CpG names and ensure they match the Meth_M info
length(which(names_typeII_sbeSNP %in% rownames(Meth_M)))  #only
6920, other 69 may be control probes
keep_typeII_sbeSNP <- typeII_sbeSNP[which(names_typeII_sbeSNP %in%
rownames(Meth_M)),]
dim(keep_typeII_sbeSNP)  #6920  33
keep_names_typeII_sbeSNP <-
rownames(typeII_sbeSNP[which(names_typeII_sbeSNP %in%
rownames(Meth_M)),])
length(keep_names_typeII_sbeSNP)
# check
identical(rownames(keep_typeII_sbeSNP), keep_names_typeII_sbeSNP)
#TRUE, good

# Pull out these CpGs from the Meth_M data
Meth_M_sbeSNP <- Meth_M[keep_names_typeII_sbeSNP,]
dim(Meth_M_sbeSNP)  # 6920   47
identical(keep_names_typeII_sbeSNP,rownames(Meth_M_sbeSNP))  #TRUE

# Order by Standard Deviation
SD_Meth_M_sbeSNP <- as.matrix(apply(Meth_M_sbeSNP,1,sd))
ord_SD_Meth_M_sbeSNP <-
SD_Meth_M_sbeSNP[rev(order(SD_Meth_M_sbeSNP)),]
Top100_SD_Meth_M_sbeSNP <- as.matrix(ord_SD_Meth_M_sbeSNP[1:100])
par(mfrow=c(3,4))
plotCpg(Meth_B, rownames(Top100_SD_Meth_M_sbeSNP)[1:20],
pheno=Meth_info@phenoData@data$Family, measure="beta", ylim= c(0,1))
# the pheno is wrong here, need to match up with the correct data
frame but for these purposes it doesn't matter as I don't currently
care how they segregate, just as long as there are 3 groups
# Also have plotted as B than M as this is more biologically
interpretable
# These look quite variable.  There are some that appear randomly
vriable though, ie. may need to perfect the kmeans cluster method to
get three nice clusters rather than just random variation

# Now go back and pull the top 100 variable out of the 6920 sbeSNP
top100sbeSNP <-
keep_typeII_sbeSNP[rownames(Top100_SD_Meth_M_sbeSNP),]
# check
top100sbeSNP$Type  # yes all II
top100sbeSNP$pos
```

```
top100sbeSNP$chr  # yes all diff chr

# Now pull the actual methylation values to use in the association
model
top100Meth_M <- Meth_M[rownames(Top100_SD_Meth_M_sbeSNP),]
# check
identical(rownames(top100Meth_M), rownames(top100sbeSNP)) # TRUE
# Subset Meth_B for biologically relevant values and to use in the
assoication model if the top100Meth_M does not work as has been a
problem in the past because low values generate NaNs
top100Meth_B <- Meth_B[rownames(Top100_SD_Meth_M_sbeSNP),]
# Use top100sbeSNP as the object for annotation info and
top100Meth_M or top100Meth_B as the actual methylation data for the
model




########################
## 2. K-means Cluster ##
########################

# This method was not as efficient as SD or 95%-range as it requires
extensive optimisation of the parameters for within and between sum
of squares so clusters are generated in three regions. For example
adjustment of within ss cut offs indicate which samples are to be
included in the cluster and adjustment of between ss indicates the
distance between clusters, optinmally clustered in three distinct
groups, not random variation across the possible beta value range of
0:1.  If not performed in parallel the function is also
computationally slow and requires the data to be broken into to
chunks

# for examples trying a totss >1 and withinss <0.15 was much more
effective than default metrics but still required further
optimisation for similar efficacy as SD or 95%-range

# totss >1  and withinss <0.15
Meth_M=t(Meth_M)
betweenSS1=vector()
modelnum=1
for(i in colnames(Meth_M)){
  cluster=kmeans(Meth_M[,i],3)
  betweenSS1[modelnum]=if(cluster$totss>1 && cluster$withinss<0.15)
cluster$betweenss/cluster$totss else "NA"
  modelnum= modelnum +1 }




############################
## 3. 95%-reference range ##
############################
```

```
# The difference between the most and least methylated individuals, among 95% of the individuals forming the central distribution of methylation values.  This approach is less sensitive to outliers than the full range and more readily interpretable than SD

dim(Meth_M) # [1] 467263     47
quantile(Meth_M[1,], probs=c(0, .025, 0.5, 0.975, 1))  # Test the quantile function on one CpG site, dividing the distribution into quintiles and selecting the lower 2.5% and upper 97.5% as the 2nd and 4th quantiles
      # 0%     2.5%      50%    97.5%     100%
# 2.096701 2.211459 2.819828 3.769314 3.914120
quantile_95_M <- apply(Meth_M,1, quantile, probs=c(0, .025, 0.5, 0.975, 1))  # apply to all CpG sites
range_95_M <- as.matrix(quantile_95_M[4,]-quantile_95_M[2,]) # calculate the 95%-range for each CpG site by subtracting the 2nd quantile from the 4th
ord_range_95 <- range_95_M[order(range_95_M, decreasing=TRUE),] # order the 95% range in decreasing order so those CpGs with the greatest range are first
Top100_range_M <- as.matrix(ord_range_95_M[1:100]) # select the first 100 CpG sites with the greatest 95%-range
Top500_range_M <- as.matrix(ord_range_95_M[1:500]) # select the first 500 CpG sites with the greatest 95%-range

## Top 100/500 variable by 95% reference range, only sbe SNPs, no probe SNPs
quantile_95_M_sbeSNP <- apply(Meth_M_sbeSNP,1, quantile, probs=c(0, .025, 0.5, 0.975, 1))
range_95_M_sbeSNP <- as.matrix(quantile_95_M_sbeSNP[4,]-quantile_95_M_sbeSNP[2,])
ord_range_95_M_sbeSNP <- range_95_M_sbeSNP[order(range_95_M_sbeSNP, decreasing=TRUE),]
Top100_range_M_sbeSNP <- as.matrix(ord_range_95_M_sbeSNP[1:100])
Top500_range_M_sbeSNP <- as.matrix(ord_range_95_M_sbeSNP[1:500])

length(which(rownames(Top100_range_M) %in% rownames(Top100_range_M_sbeSNP))) # 90, 90%
# Do association analysis for Top100_range_M because 90/100 of them were sbe SNPs so wanted to know what the other 10 looked like, ie do they have lower log p-vals like I suspect
# Use Top100_range_M in the model, if Beta values are required used the CpG names from this object to pull the correct Beta values




#########################
## Perform Association ##
#########################

## Data required for both analysis ##
# a) Selecting samples
colnames(top100Meth_M)  # good quality samples on methylation array
```

```
 # [1] "PC11.3"   "PC22.2"   "PC11.4"   "PC22.3"   "PC11.9"
 # [6] "PC22.16"  "PC11.147" "PC22.17"  "PC22.21"  "PC22.468"
# [11] "PC22.203" "PC22.387" "PC72.136" "PC22.416" "PC72.188"
# [16] "PC72.4"   "PC72.213" "PC72.77"  "PC72.126" "PC9.1"
# [21] "PC9.4"    "PC9.12"   "PC9.24"   "PC9.121"  "PC9.477"
# [26] "PC22.210" "PC22.393" "PC22.414" "PC22.4"   "PC22.162"
# [31] "PC22.195" "PC11.415" "PC22.418" "PC11.234" "PC22.476"
# [36] "PC11.180" "PC72.03"  "PC22.388" "PC9.338"  "PC22.386"
# [41] "PC9.357"  "PC22.274" "PC9.129"  "PC72.106" "PC9.29"
# [46] "PC72.187" "PC9.286"
# create a text file with these samples so they can be pulled from
the plink file
# Create a fam file with all the samples from the genotyping data
then match these to the ones that have methylation data.  NB, have
used the position of the first variable CpG site plus 1Mb to make it
fast as I don't need all the information.
system("/Applications/plink-1.07-mac-intel/plink --noweb --bfile /
Users/ecazaly/Desktop/PhD_Analysis/Association_2015april/
Ass_genetic_input/cutoff_15_final --chr 1 --from-bp 33292126 --to-bp
34292126 --transpose --recode --out /Users/ecazaly/Desktop/
PhD_Analysis/Association_2015april/Perform_Ass/Top100variable/
geno_sampleNames")
# Use the .tfam file generated by this command to look at samples
geno_samples <- read.table("geno_sampleNames.tfam")  # sample names
are V2 column. There are 51 samples
# change the format of methylation names so they can be compared
topVariable_samples <-  colnames(top100Meth_M)
topVariable_samples <- gsub(".", "_",topVariable_samples, fixed=T)
topVariable_samples[20] <- gsub("_1", "_01",topVariable_samples[20],
fixed=T)
topVariable_samples[2] <- gsub("_2", "_02",topVariable_samples[2],
fixed=T)
topVariable_samples[c(1,4)] <- gsub("_3",
"_03",topVariable_samples[c(1,4)], fixed=T)
topVariable_samples[c(3,16,21,29)] <- gsub("_4",
"_04",topVariable_samples[c(3,16,21,29)], fixed=T)
topVariable_samples[5] <- gsub("_9", "_09",topVariable_samples[5],
fixed=TRUE)
topVariable_samples[16] <- gsub("04","04_a",
topVariable_samples[16], fixed=TRUE)
topVariable_samples[8] <- gsub("17","17_a", topVariable_samples[8],
fixed=TRUE)
write.table(topVariable_samples, file="meth_samplesNames.txt",
col.names=FALSE, quote=FALSE)
meth_samplesNames <- read.table("meth_samplesNames.txt")
# match the two sample files to see which to keep
length(geno_samples$V2)  #51
length(meth_samplesNames$V2) #47
length(which(geno_samples$V2 %in% meth_samplesNames$V2)) #39, agrees
with previous scripts
# create new text file with just these names to pull out desired
samples from plink
keep_sampleNames <- geno_samples[which(geno_samples$V2 %in%
meth_samplesNames$V2),]
```

```
# make sure keep V1 as this is the family ID.  I didn't put a proper
family ID in the genotyping file so it's just a number 1-51
write.table(keep_sampleNames[,1:2], col.names=FALSE,
row.names=FALSE, quote=FALSE, file="keep_samples.txt")

# b) creating a kinship coefficient matrix to include how samples
are related in the association model
# Use the ibs (identity by state) function from the GenABEL package.
Here I've created the ibs_no object by choosing "no" for the weight
argument in the ibs function, this allows direct IBS computaitons.
I tested an alternative method, "weight=freq" which takes into
account allelic frequency assuming HWE but this method was not
sucessful in the association model, possibly because the assumption
did not hold for my familial data where the allelic frequencies may
not be in HWE
gwaa_all_omni=load.gwaa.data(phenofile="/Users/ecazaly/Desktop/
PhD_Analysis/Association_2015april/Ass_genetic_input/SBE_snp/
pheno_cg13387643.txt", genofile="/Users/ecazaly/Desktop/
PhD_Analysis/Association_2015april/Ass_genetic_input/SBE_snp/
all_omni.raw", force=TRUE)
ibs_no <- ibs(gwaa_all_omni, weight="no")
colnames(ibs_no) <- gsub("_", "-", colnames(ibs_no))
rownames(ibs_no) <- gsub("_", "-", rownames(ibs_no))


####################################
## 1. Standard Deviation Method  ##
####################################
# Use top 100 variable as determined by SD with no probe SNPs and
all sbe SNPs in Type II probes
# Export genotype data from PLINK and create R object with 250Kb
window, 125Kb (50^3) either side (tested 200kb/400kb/500kb/1Mb,
there was only 1-5 snps in the 200Kb-400Kb for the first cpg but
increasing to 2MB gave 704 SNPs in the 1st Cpg window but some other
cpgs had far too many at a couple of thousand)

library(GenABEL)
library(hglm)
chr <- top100sbeSNP$chr
chr <- as.integer(gsub("chr", "", chr))
pos <- top100sbeSNP$pos
cpgNames <- rownames(top100sbeSNP)
cpg <- 1:length(chr)

for(cpg in 1:length(chr)){
system(paste(c("/Applications/plink-1.07-mac-intel/plink --noweb --
bfile /Users/ecazaly/Desktop/PhD_Analysis/Association_2015april/
Ass_genetic_input/cutoff_15_final --keep /Users/ecazaly/Desktop/
PhD_Analysis/Association_2015april/Perform_Ass/Top100variable/
keep_samples.txt --chr ", chr[cpg]," --from-bp ",pos[cpg]-50^3," --
to-bp ",pos[cpg]+50^3," --recode --transpose --out /Users/ecazaly/
Desktop/PhD_Analysis/Association_2015april/Perform_Ass/
Top100variable/",cpgNames[cpg]),collapse=""))
} # Files should be in .tped and .tmap format for the GenABEL R
```

```
package;  add in --transpose line

# Create genotype .raw file
library(GenABEL)
cpg <- 1:length(cpgNames)
for(cpg in 1:length(cpgNames)){
convert.snp.tped(tpedfile=paste("/Users/ecazaly/Desktop/
PhD_Analysis/Association_2015april/Perform_Ass/
Top100variable/",cpgNames[cpg],".tped", sep=""), tfamfile=paste("/
Users/ecazaly/Desktop/PhD_Analysis/Association_2015april/
Perform_Ass/Top100variable/",cpgNames[cpg],".tfam", sep=""),
outfile=paste("/Users/ecazaly/Desktop/PhD_Analysis/
Association_2015april/Perform_Ass/
Top100variable/",cpgNames[cpg],".raw", sep=""))
}
# Create phenotype file from 'top100Meth_B' data.  Tried
top100Meth_M but this fails in the model
colnames(top100Meth_B) <- topVariable_samples
length(which(colnames(top100Meth_B) %in% keep_sampleNames$V2))
keep_top100Meth_B <-top100Meth_B[,which(colnames(top100Meth_B) %in%
keep_sampleNames$V2)]
dim(keep_top100Meth_B)
colnames(keep_top100Meth_B)
ID <- matrix(colnames(keep_top100Meth_B))
colnames(ID) <- "id"
female <- keep_sampleNames[keep_sampleNames$V5==2,]
female$V2
sex <- matrix(c(rep(1,17),0,1,1,0,0,0,rep(1,11),0,1,0,0,1))
colnames(sex) <- "sex"
pheno <- cbind(ID, sex, t(keep_top100Meth_B))
write.table(pheno, file="/Users/ecazaly/Desktop/PhD_Analysis/
Association_2015april/Perform_Ass/Top100variable/
pheno.txt",quote=FALSE)

for(cpg in 1:length(cpgNames)){
        require(GenABEL)
gwaa=load.gwaa.data(phenofile="/Users/ecazaly/Desktop/PhD_Analysis/
Association_2015april/Perform_Ass/Top100variable/pheno.txt",
genofile=paste("/Users/ecazaly/Desktop/PhD_Analysis/
Association_2015april/Perform_Ass/Top100variable/",cpgInfo[cpg,
3],".raw", sep=""),  force=TRUE)
gt=as.data.frame(as.numeric(gtdata(gwaa)))
gwaa@gtdata@gtps=gt
gwaa@phdata$id=gsub("_","-",gwaa@phdata$id)
rownames(gwaa@phdata)=gsub("_", "-", rownames(gwaa@phdata))
gwaa@gtdata@idnames=gsub("_", "-", gwaa@gtdata@idnames)
rownames(gwaa@gtdata@gtps)=gsub("_", "-",
rownames(gwaa@gtdata@gtps))
modelnum=1
name=vector()
coefs=vector()
pvals=vector()
log10pvps=vector()
Number_Sig_Hits=vector()
```

```
Highest_pval=vector()
details=matrix(nrow=1, ncol=2)
colnames(details)=c("Number_Sig_Hits","Highest_pval")
rownames(details)=cpgInfo[cpg,3]
for(i in colnames(gwaa@phdata[(cpgInfo[cpg,4])+2])) {
  for(j in colnames(gwaa@gtdata@gtps)){
        if(sum(gwaa@gtdata@gtps[j], na.rm=T)<1) {
                next  }
        name[modelnum]= paste(i, j, sep= "/")
    formula=as.formula(paste(gwaa@phdata[i], "~",
gwaa@gtdata@gtps[j]))
    association=polygenic_hglm(formula, ibs_no, gwaa)
    coefs[modelnum]= summary(association$hglm)$FixCoefMat[2,1]
    pvals[modelnum]= summary(association$hglm)$FixCoefMat[2,4]
    log10pvps[modelnum]= -1*log10(pvals[modelnum])
    modelnum= modelnum + 1
    }}
#plot
system(paste("/Applications/plink-1.07-mac-intel/plink --noweb --
bfile /Users/ecazaly/Desktop/PhD_Analysis/Association_2015april/
Ass_genetic_input/cutoff_15_final --keep /Users/ecazaly/Desktop/
PhD_Analysis/Association_2015april/Perform_Ass/Top100variable/
keep_samples.txt --chr ", cpgInfo[cpg,1]," --from-bp ",cpgInfo[cpg,
2]-50^3," --to-bp ",cpgInfo[cpg,2]+50^3," --recode --out /Users/
ecazaly/Desktop/PhD_Analysis/Association_2015april/Perform_Ass/
Top100variable/", cpgInfo[cpg,3], sep=""))
 # don't transpose to get .map file
map <- read.table(paste("/Users/ecazaly/Desktop/PhD_Analysis/
Association_2015april/Perform_Ass/Top100variable/",cpgInfo[cpg,
3],".map",sep=""), as.is=T)
# cut down map to the snps that have pvals
name_keep <- gsub(paste(cpgInfo[cpg,3], "/", sep=""),"", name,
fixed=TRUE)
map_b <- map[c(which(name_keep %in% map$V2)),]
nsnps <- nrow(map_b)
png(filename=paste("PNGs_IBS_no/", cpgInfo[cpg,
3],"_ibs",".png",sep=""),pointsize=12,units="mm",width=137.6,height=
137.6*2/3,res=800)
        par(family="serif")
        par(mar=c(4,4,3,0.5))
        plot(map_b[,4]*10^-6,
log10pvps,type="o",main=NULL,xlab=c("Position (Mb)",
cex=2),ylab="-1*log10(p-value)",xlim=c(map_b[1,4]*10^-6,map_b[nsnps,
4]*10^-6),ylim=c(-3,35),cex=0.2)
        title(main=list(paste("Association between ",cpgInfo[cpg,
3], " and ", nsnps, " SNPs in a 250Kb window", sep=""), cex=.95))
        abline(h=-1*log10(0.05),lty=2)
        text(map_b[nsnps,4]*10^-6-0.005,-1*log10(0.05)+1,"p-value =
0.05",font=3, cex=0.4, col="red")
        abline(h=-1*log10(0.05/nsnps),lty=2)
        text(map_b[nsnps,4]*10^-6-0.005,-1*log10(0.05/nsnps)
+1.2,"adjusted p-value",font=3, cex=0.4, col="red")
        text(map_b[nsnps,4]*10^-6-0.033, 35, paste("Significant
associations: ", length(which(log10pvps> -1*log10(0.05/nsnps))),
```

```
spe=""), cex=0.6)
        dev.off()
# significant hit info
Number_Sig_Hits=length(which(log10pvps>-1*log10(0.05/nsnps)))
Highest_pval=max(log10pvps, na.rm=T)
details[,1]=Number_Sig_Hits
details[,2]=Highest_pval
write.csv(details, file=paste("/Users/ecazaly/Desktop/PhD_Analysis/
Association_2015april/Perform_Ass/Top100variable/Hits_ibs_no/",
cpgInfo[cpg,3], ".csv", sep=""))
}
# then read in and bind all csvs
filenames <- list.files(path="/Users/ecazaly/Desktop/PhD_Analysis/
Association_2015april/Perform_Ass/Top100variable/Hits_ibs_no/",
full.names=T)
library(plyr)
import.list <- llply(filenames, read.csv, header=T)
details_all <- matrix(nrow=100, ncol=3)
library(reshape)
details_all <- merge_recurse(import.list)
colnames(details_all) <- c("CpG_Name", "Number_Sig_Hits",
"Highest_pval")
write.csv(details_all, file="hits_ibs_no.csv")
hits_ibs_no <- read.csv("hits_ibs_no.csv")

# then add annotation info using the minfi package and it's add ons
Genomic_Position <- cpgInfo[,2]
Chromosome <- cpgInfo[,1]
Associated_gene <- top100sbeSNP[,"UCSC_RefGene_Name"]
rs_name <- top100sbeSNP[,"SBE_rs"]
CpG_location <- top100sbeSNP[,"Relation_to_Island"]
MAF <- top100sbeSNP[,"SBE_maf"] # for these it is the same as
top100sbeSNP[,"CpG_maf"]
identical(top100sbeSNP[,"SBE_maf"], top100sbeSNP[,"CpG_maf"])  #TRUE

CPG_detail_100 <- matrix(nrow=100, ncol=11)
row.names(CPG_detail_100)=cpgNames
colnames(CPG_detail_100)=COLnames
CPG_detail_100[,1] <- Genomic_Position
CPG_detail_100[,2] <- Chromosome
CPG_detail_100[,5] <- Associated_gene
CPG_detail_100[,7] <- rs_name
CPG_detail_100[,8] <- MAF
rs_onOMNI_100 <- conversion[conversion$RsID %in% CPG_detail_100[,
7],]
for(i in rownames(CPG_detail_100)){
CPG_detail_100[i,9] <- if(CPG_detail_100[i,7] %in%
rs_onOMNI_100$RsID) "yes" else "no"
}
CPG_detail_100[,10] <- CpG_location
write.csv(CPG_detail_100, file="CPG_detail_100.csv")
```

```
########################
## 2. 95%-range Method ##
########################
# make new cpgInfo file
cpgNames_range <- rownames(Top100_range_M)
Annotated_range <- Annotated_meth[cpgNames_range,]
identical(rownames(Meth_M_100range), rownames(Annotated_range))
#TRUE
Meth_M_100range <- Meth_M[cpgNames_range,]
identical(rownames(Annotated_range), cpgNames_range)  #TRUE
identical(rownames(Meth_M_100range), cpgNames_range)  #TRUE
chr_range <- Annotated_range$chr
chr_range <- as.integer(gsub("chr", "", chr_range))
pos_range <- Annotated_range$pos
cpgNumber_range <- 1:length(chr_range)
cpgInfo_range <- data.frame(chr_range, pos_range, cpgNames_range,
cpgNumber_range)
cpgInfo_range$cpgNames_range <- as.character(cpgInfo_range
$cpgNames_range)

# Create .tped files
for(cpg in 1:length(chr_range)){
system(paste(c("/Applications/plink-1.07-mac-intel/plink --noweb --
bfile /Users/ecazaly/Desktop/PhD_Analysis/Association_2015april/
Ass_genetic_input/cutoff_15_final --keep /Users/ecazaly/Desktop/
PhD_Analysis/Association_2015april/Perform_Ass/Top100variable/
keep_samples.txt --chr ", chr_range[cpg]," --from-bp
",pos_range[cpg]-50^3," --to-bp ",pos_range[cpg]+50^3," --recode --
transpose --out /Users/ecazaly/Desktop/PhD_Analysis/
Association_2015april/Perform_Ass/Top100_range/",
cpgNames_range[cpg]),collapse=""))}
# Create .raw file
library(GenABEL)
for(cpg in 1:length(cpgNames_range)){
convert.snp.tped(tpedfile=paste("/Users/ecazaly/Desktop/
PhD_Analysis/Association_2015april/Perform_Ass/Top100_range/",
cpgNames_range[cpg],".tped", sep=""), tfamfile=paste("/Users/
ecazaly/Desktop/PhD_Analysis/Association_2015april/Perform_Ass/
Top100_range/", cpgNames_range[cpg],".tfam", sep=""),
outfile=paste("/Users/ecazaly/Desktop/PhD_Analysis/
Association_2015april/Perform_Ass/Top100_range/",
cpgNames_range[cpg],".raw", sep=""))}
# Create phenotype file, try with Meth_M
Meth_M_100range
# select only the samples that have geno data for
colnames(Meth_M_100range) <- colnames(top100Meth_B)
length(which(colnames(Meth_M_100range) %in% keep_sampleNames$V2))
#39, keep the samples which have both methylation and Omni2.5 SNP
genotyping data
keep_Meth_M_100range <-
Meth_M_100range[,which(colnames(Meth_M_100range) %in%
keep_sampleNames$V2)]
dim(keep_Meth_M_100range)
ID <- matrix(colnames(keep_Meth_M_100range))
```

```
colnames(ID) <- "id"
female <- keep_sampleNames[keep_sampleNames$V5==2,]
female$V2
sex <- matrix(c(rep(1,17),0,1,1,0,0,0,rep(1,11),0,1,0,0,1))
colnames(sex) <- "sex"
pheno_range <- cbind(ID, sex, t(keep_Meth_M_100range))
write.table(pheno_range, file="/Users/ecazaly/Desktop/PhD_Analysis/
Association_2015april/Perform_Ass/Top100_range/
pheno.txt",quote=FALSE)
# the association model did not work with these M-values as the
pheno input, try with Beta values and see if that works
Meth_B_100range <- Meth_B[cpgNames_range,]
colnames(Meth_B_100range) <- colnames(top100Meth_B)
length(which(colnames(Meth_B_100range) %in% keep_sampleNames$V2))
#39
keep_Meth_B_100range <-
Meth_B_100range[,which(colnames(Meth_B_100range) %in%
keep_sampleNames$V2)]
ID <- matrix(colnames(keep_Meth_B_100range))
colnames(ID) <- "id"
female <- keep_sampleNames[keep_sampleNames$V5==2,]
female$V2
sex <- matrix(c(rep(1,17),0,1,1,0,0,0,rep(1,11),0,1,0,0,1))
colnames(sex) <- "sex"
pheno_range_B <- cbind(ID, sex, t(keep_Meth_B_100range))
write.table(pheno_range_B, file="/Users/ecazaly/Desktop/
PhD_Analysis/Association_2015april/Perform_Ass/Top100_range/
pheno_B.txt",quote=FALSE)


# Association
for(cpg in 1:length(cpgNames_range)){
        require(GenABEL)
gwaa=load.gwaa.data(phenofile="/Users/ecazaly/Desktop/PhD_Analysis/
Association_2015april/Perform_Ass/Top100_range/pheno_B.txt",
genofile=paste("/Users/ecazaly/Desktop/PhD_Analysis/
Association_2015april/Perform_Ass/Top100_range/", cpgInfo_range[cpg,
3],".raw", sep=""),  force=TRUE)
gt=as.data.frame(as.numeric(gtdata(gwaa)))
gwaa@gtdata@gtps=gt
gwaa@phdata$id=gsub("_","-",gwaa@phdata$id)
rownames(gwaa@phdata)=gsub("_", "-", rownames(gwaa@phdata))
gwaa@gtdata@idnames=gsub("_", "-", gwaa@gtdata@idnames)
rownames(gwaa@gtdata@gtps)=gsub("_", "-",
rownames(gwaa@gtdata@gtps))
modelnum=1
name=vector()
coefs=vector()
pvals=vector()
log10pvps=vector()
Number_Sig_Hits=vector()
Highest_pval=vector()
details_range=matrix(nrow=1, ncol=2)
colnames(details_range)=c("Number_Sig_Hits","Highest_pval")
```

```
rownames(details_range)=cpgInfo_range[cpg,3]

for(i in colnames(gwaa@phdata[(cpgInfo_range[cpg,4])+2])) {
  for(j in colnames(gwaa@gtdata@gtps)){
        if(sum(gwaa@gtdata@gtps[j], na.rm=T)<1) {
                next  }
        name[modelnum]= paste(i, j, sep= "/")
    formula=as.formula(paste(gwaa@phdata[i], "~",
gwaa@gtdata@gtps[j]))
    association=polygenic_hglm(formula, ibs_no, gwaa)
    coefs[modelnum]= summary(association$hglm)$FixCoefMat[2,1]
    pvals[modelnum]= summary(association$hglm)$FixCoefMat[2,4]
    log10pvps[modelnum]= -1*log10(pvals[modelnum])
    modelnum= modelnum + 1
    }}
# plot
 system(paste("/Applications/plink-1.07-mac-intel/plink --noweb --
bfile /Users/ecazaly/Desktop/PhD_Analysis/Association_2015april/
Ass_genetic_input/cutoff_15_final --keep /Users/ecazaly/Desktop/
PhD_Analysis/Association_2015april/Perform_Ass/Top100variable/
keep_samples.txt --chr ", cpgInfo_range[cpg,1]," --from-bp
",cpgInfo_range[cpg,2]-50^3," --to-bp ",cpgInfo_range[cpg,2]+50^3,"
--recode --out /Users/ecazaly/Desktop/PhD_Analysis/
Association_2015april/Perform_Ass/Top100_range/", cpgInfo_range[cpg,
3], sep=""))
 # don't transpose to get .map file
map <- read.table(paste("/Users/ecazaly/Desktop/PhD_Analysis/
Association_2015april/Perform_Ass/Top100_range/", cpgInfo_range[cpg,
3],".map",sep=""), as.is=T)
# cut down map to the snps that have pvals
name_keep <- gsub(paste(cpgInfo_range[cpg,3], "/", sep=""),"", name,
fixed=TRUE)
map_b <- map[c(which(name_keep %in% map$V2)),]
nsnps <- nrow(map_b)

png(filename=paste("/Users/ecazaly/Desktop/PhD_Analysis/
Association_2015april/Perform_Ass/Top100_range/PNGs_range/",
cpgInfo_range[cpg,
3],".png",sep=""),pointsize=12,units="mm",width=137.6,height=137.6*2
/3,res=800)
        par(family="serif")
        par(mar=c(4,4,3,0.5))
        plot(map_b[,4]*10^-6,
log10pvps,type="o",main=NULL,xlab=c("Position (Mb)",
cex=2),ylab="-1*log10(p-value)",xlim=c(map_b[1,4]*10^-6,map_b[nsnps,
4]*10^-6),ylim=c(-3,35),cex=0.2)
        title(main=list(paste("Association between
",cpgInfo_range[cpg,3], " and ", nsnps, " SNPs in a 250Kb window",
sep=""), cex=.95))
        abline(h=-1*log10(0.05),lty=2)
        text(map_b[nsnps,4]*10^-6-0.005,-1*log10(0.05)+1,"p-value =
0.05",font=3, cex=0.4, col="red")
        abline(h=-1*log10(0.05/nsnps),lty=2)
        text(map_b[nsnps,4]*10^-6-0.005,-1*log10(0.05/nsnps)
```

```
+1.2,"adjusted p-value",font=3, cex=0.4, col="red")
        text(map_b[nsnps,4]*10^-6-0.033, 35, paste("Significant
associations: ", length(which(log10pvps> -1*log10(0.05/nsnps))),
spe=""), cex=0.6)
        dev.off()

Number_Sig_Hits=length(which(log10pvps>-1*log10(0.05/nsnps)))
Highest_pval=max(log10pvps, na.rm=T)
details_range[,1]=Number_Sig_Hits
details_range[,2]=Highest_pval
write.csv(details_range, file=paste("/Users/ecazaly/Desktop/
PhD_Analysis/Association_2015april/Perform_Ass/Top100_range/
Hits_range/", cpgInfo_range[cpg,3], ".csv", sep=""))
}

# then read in and bind all csvs
filenames <- list.files(path="/Users/ecazaly/Desktop/PhD_Analysis/
Association_2015april/Perform_Ass/Top100_range/Hits_range/",
full.names=T)
library(plyr)
import.list <- llply(filenames, read.csv, header=T)
details_all_range <- matrix(nrow=100, ncol=3)
library(reshape)
details_all_range <- merge_recurse(import.list)
colnames(details_all_range) <- c("CpG_Name", "Number_Sig_Hits",
"Highest_pval")
write.csv(details_all_range, file="/Users/ecazaly/Desktop/
PhD_Analysis/Association_2015april/Perform_Ass/Top100_range/
all_hits_range.csv")

#get info on the hits
range_sigHits_annotated <- Annotated_meth[details_all_range
$CpG_Name,]
dim(range_sigHits_annotated)  # [1] 98  33
# 96 sig hits from the 98 ass that worked, 85 of these are >10log
pval
# check these for CpG_rs
range_10log <- details_all_range[which(details_all_range
$Highest_pval>=10),]
dim(range_10log)  #85  3
range_10log_cpgs <- range_10log$CpG_Name
range_10log_annotated <- Annotated_meth[range_10log_cpgs,
c(1,2,4,9,12,14,15,16,17,18,19,24,28,29,33)]
range_10log_annotated$CpG_rs  # all but 4 have CpG_rs
length(which(!(is.na(range_10log_annotated$CpG_rs))))  #81
length(which(!(is.na(range_sigHits_annotated$CpG_rs))))  #90

# check overlap with SD
length(which(rownames(range_sigHits_annotated) %in%
rownames(Top100_SD_M))) #66
rownames(range_sigHits_annotated[rownames(range_sigHits_annotated)
%in% rownames(Top100_SD_M),])
length(which(rownames(range_sigHits_annotated) %in%
rownames(Top500_SD_M))) #98
```

```
# write range signifcant hits to a csv with annotaion details and
log pvals etc to be used to prioritising CpGs for validation and
follow up
range_sigHits_detail <-
range_sigHits_annotated[,c(1,2,4,9,10,12,14,15,16,19,24,29)]
head(range_sigHits_detail)
identical(as.character(details_all_range$CpG_Name),
rownames(range_sigHits_detail)) # TRUE
range_sigHits_detail$num_hits <- details_all_range$Number_Sig_Hits
range_sigHits_detail$highest_pval <- details_all_range$Highest_pval
write.csv(range_sigHits_detail, file="/Users/ecazaly/Desktop/
PhD_Analysis/Association_2015april/Ass_genetic_input/
Genetic_riskLoci/Hits_combined/range_sigHits_detail.csv")

# Use the csv files with genome annotations and log p-vals to
prioritise CpGs for validation and further follow up.
```

```
#################################################################
##  Prostate Cancer Risk Loci Approach to Identify meQ 3QTLs ##
#################################################################


# generate prostate cancer rick genomic windows to examine the
association between SNPs in the region and methylation

# For our linkage regions and GWAS, a 50KB window was taken around
the hit SNP.  For published linkage regions a sliding window
approach was used as described below

# Sliding windows were created by selecting a core region of 30KB
with an additional 10KB either side which overlaps with wondows
either side. This size window was chosen as (Smith et al 2014, Zhi
et al 2013) have found 15KB/10KB to be average distance for
association, this gives a little extra incase.  Larger windows were
not chosen as (Luijk et al 2015) have suggested that windows of
hundreds of KB are much too large due to high FDR, they suggest
10-50KB.  These windows are much less than the 250KB windows used in
the methylation approach as that method examines only one CpG per
window and thus has a lower number of tests and FDR.
# if ROI are greater than 50KB then take a few 50KB windows with an
overlap of 10KB each side, if less than 50KB then take whole region
and add however many KBs to each side to make the total window 50KB


library(minfi)
library(GenABEL)
library(doParallel)
library(foreach)
numCores <- detectCores()-1
cl <- makeCluster(numCores)
registerDoParallel(cl)

## Data required for all analysis ##
# the same files/R objects generated for the methylation apparoach
were used to:
        # a) Select samples
        # b) Kinship coefficient matrix



#####################################################################
#########################
## 1. Familial Prostate Cancer risk regions identified in our lab
through linkage analysis ##
#####################################################################
#########################

# In a familial prostate cancer linkage analysis performed in our
lab (REFERENCE), 4 regions with high LOD scores were identified as
possible prostate cancer risk regions. Within these regions 143 SNPs
were identified (ask Nick about exactly how he did this) on four
```

```
chromosomes (2,6,15,22)
# The regions were then converted to the hg19 genome annotation via
SNPnexus


PCrisk_list <- read.csv(file="/Users/ecazaly/Desktop/PhD_Analysis/
Association_2015april/Perform_Ass/PCrisk/
genomic_riskLoci_Leisel_geneNames.csv", header=T)
PCrisk_list$Liesel_LinkageROI_List # look at genes
genes_Leisel <- annotated_genes[which(annotated_genes
$UCSC_RefGene_Name %in% PCrisk_list$Liesel_LinkageROI_List),]
dim(genes_Leisel)
# [1] 1428   33
genes_Leisel$UCSC_RefGene_Name  # gene names, there are duplicates
as some genes have more than one cpg
genes_Leisel$Name  # these are the cpg sites
# pull these from Meth_B
genes_Leisel_Meth_B <- Meth_B[which(rownames(Meth_B) %in%
genes_Leisel$Name),]
dim(genes_Leisel_Meth_B)
# [1] 1427  47
genes_Leisel_info <- Meth_info[genes_Leisel$Name,
genes_Leisel_info_annotation <-getAnnotation(genes_Leisel_info)
check_gene_Leisel_Names <- genes_Leisel_info_annotation[genes_Leisel
$Name, "UCSC_RefGene_Name"]
length(check_gene_Leisel_Names %in% PCrisk_list
$Liesel_LinkageROI_List) # 1428
# any of these CpGs in the TopVariable?
which(genes_Leisel$Name %in% rownames(top100sbeSNP))  #1:  1298
# Top 500 ?
which(rownames(Top500_SD_Meth_M_sbeSNP) %in% genes_Leisel$Name) #
same 1 1298 / 71
Top500_SD_Meth_M_sbeSNP[71,]
genes_Leisel[1298,]
# this is cg20205188 the one that was pulled out at the top of the
script, no more by looking at top 500.
# what about looking at CpGs within the region.. may not necessarily
be associated with the gene..
# but also positions themselves
chrAll_risk$chr
chrAll_risk$pos # maybe pull cpgs within range?
chrAll_risk$chrPos <- paste(chrAll_risk$chr, ":", chrAll_risk$pos,
sep="")
colnames(chrAll_risk)
write.csv(chrAll_risk, file="Leisel_list.csv")


leisel <- read.csv(file="/Users/ecazaly/Desktop/PhD_Analysis/
Association_2015april/Perform_Ass/PCrisk/Leisel_risk/
Leisel_list.csv")
leisel <- leisel[,c(2,5,6,7)]
colnames(leisel) <- c("snp", "chr", "pos", "LOD")
leisel$start_50KB <- leisel$pos-25000
leisel$end_50KB <- leisel$pos+25000
```

```
leisel$end_50KB - leisel$start_50KB # check
leisel$chr <- as.numeric(gsub("chr", "", leisel$chr, fixed=T))
# no Xchr to remove
length(leisel$snp)  #143


foreach(i=1:length(leisel$snp)) %dopar% {
system(paste(c("/Applications/plink-1.07-mac-intel/plink --noweb --
bfile /Users/ecazaly/Desktop/PhD_Analysis/Association_2015april/
Ass_genetic_input/cutoff_15_final --keep /Users/ecazaly/Desktop/
PhD_Analysis/Association_2015april/Perform_Ass/Top100variable/
keep_samples.txt --chr ", leisel$chr[i]," --from-bp ", leisel
$start_50KB[i]," --to-bp ", leisel$end_50KB[i]," --recode --
transpose --out /Users/ecazaly/Desktop/PhD_Analysis/
Association_2015april/Ass_genetic_input/Genetic_riskLoci/
windows_leisel/", as.character(leisel$snp[i])),collapse=""))
foreach(i=1:length(leisel$snp)) %dopar% {
require(GenABEL)
convert.snp.tped(tpedfile=paste("/Users/ecazaly/Desktop/
PhD_Analysis/Association_2015april/Ass_genetic_input/
Genetic_riskLoci/windows_leisel/", as.character(leisel
$snp[i]),".tped", sep=""), tfamfile=paste("/Users/ecazaly/Desktop/
PhD_Analysis/Association_2015april/Ass_genetic_input/
Genetic_riskLoci/windows_leisel/",as.character(leisel
$snp[i]),".tfam", sep=""), outfile=paste("/Users/ecazaly/Desktop/
PhD_Analysis/Association_2015april/Ass_genetic_input/
Genetic_riskLoci/windows_leisel/", as.character(leisel
$snp[i]),".raw", sep=""))
# create .map file by not transposing
system(paste(c("/Applications/plink-1.07-mac-intel/plink --noweb --
bfile /Users/ecazaly/Desktop/PhD_Analysis/Association_2015april/
Ass_genetic_input/cutoff_15_final --keep /Users/ecazaly/Desktop/
PhD_Analysis/Association_2015april/Ass_genetic_input/
keep_samples.txt --chr ", leisel$chr[i]," --from-bp ", leisel
$start_50KB[i]," --to-bp ", leisel$end_50KB[i]," --recode --out /
Users/ecazaly/Desktop/PhD_Analysis/Association_2015april/
Ass_genetic_input/Genetic_riskLoci/windows_leisel/",
as.character(leisel$snp[i])),collapse=""))
}

cpgs_leisel <- vector()
foreach(i=1:length(leisel$snp)) %dopar% {
require(minfi)
cpgs_leisel <- rownames(Annotated_meth[which(Annotated_chr==leisel
$chr[i] & Annotated_meth$pos <leisel$end_50KB[i] & Annotated_meth
$pos >leisel$start_50KB[i]),])
        cpgs_leisel <- cpgs_leisel[which(cpgs_leisel %in%
rownames(Meth_B))]
        Meth_B_leisel <- data.frame(Meth_B[cpgs_leisel,])
        Meth_B_leisel <- if(length(cpgs_leisel)==1)
t(Meth_B_leisel) else Meth_B_leisel
        colnames(Meth_B_leisel) <- topVariable_samples
        rownames(Meth_B_leisel) <- cpgs_leisel
        keep_leisel <-
```

```
data.frame(Meth_B_leisel[,which(colnames(Meth_B_leisel) %in%
keep_sampleNames$V2)])
        keep_leisel <- if(length(cpgs_leisel)==1) t(keep_leisel)
else keep_leisel
        rownames(keep_leisel) <- if(length(cpgs_leisel)==1)
cpgs_leisel else rownames(keep_leisel)
        dim(keep_leisel) # 1 39
ID <- matrix(colnames(keep_leisel))
colnames(ID) <- "id"
female <- keep_sampleNames[keep_sampleNames$V5==2,]
female$V2
sex <- matrix(c(rep(1,17),0,1,1,0,0,0,rep(1,11),0,1,0,0,1))
colnames(sex) <- "sex"
        dim(keep_leisel) # 39 1
pheno_leisel <- cbind(ID, sex, t(keep_leisel))
colnames(pheno_leisel)
write.table(pheno_leisel, file=paste("/Users/ecazaly/Desktop/
PhD_Analysis/Association_2015april/Ass_genetic_input/
Genetic_riskLoci/windows_leisel/pheno_",as.character(leisel$snp[i]),
".txt", sep=""), quote=FALSE)
}

foreach(i=1:length(leisel$snp))  %dopar% {
require(GenABEL)
gwaa=load.gwaa.data(phenofile=paste("/Users/ecazaly/Desktop/
PhD_Analysis/Association_2015april/Ass_genetic_input/
Genetic_riskLoci/windows_leisel/pheno_",as.character(leisel$snp[i]),
".txt", sep=""), genofile=paste("/Users/ecazaly/Desktop/
PhD_Analysis/Association_2015april/Ass_genetic_input/
Genetic_riskLoci/windows_leisel/", as.character(leisel
$snp[i]),".raw", sep=""),  force=TRUE)
gt=as.data.frame(as.numeric(gtdata(gwaa)))
gwaa@gtdata@gtps=gt
gwaa@phdata$id=gsub("_","-",gwaa@phdata$id)
rownames(gwaa@phdata)=gsub("_", "-", rownames(gwaa@phdata))
gwaa@gtdata@idnames=gsub("_", "-", gwaa@gtdata@idnames)
rownames(gwaa@gtdata@gtps)=gsub("_", "-",
rownames(gwaa@gtdata@gtps))
name=vector()
coefs=vector()
pvals=vector()
log10pvps=vector()
Number_Sig_Hits=vector()
Highest_pval=vector()
modelnum=1
for(cpg in colnames(gwaa@phdata[-c(1,2)])) {
  for(snp in colnames(gwaa@gtdata@gtps)){
        if(sum(gwaa@gtdata@gtps[snp], na.rm=T)<1) {
                next  }
        name[modelnum]= paste(cpg, snp, sep= "/")
    formula=as.formula(paste(gwaa@phdata[cpg], "~",
gwaa@gtdata@gtps[snp]))
    association=polygenic_hglm(formula, ibs_no, gwaa)
    coefs[modelnum]= summary(association$hglm)$FixCoefMat[2,1]
```

```
    pvals[modelnum]= summary(association$hglm)$FixCoefMat[2,4]
    log10pvps[modelnum]= -1*log10(pvals[modelnum])
    modelnum=modelnum+1
    }}
#plot
map_b <- read.table(paste("/Users/ecazaly/Desktop/PhD_Analysis/
Association_2015april/Ass_genetic_input/Genetic_riskLoci/
windows_leisel/", as.character(leisel$snp[i]),".map",sep="",
collapse=""), as.is=T)
nsnps  <- ncol(gwaa@gtdata@gtps)
nCPGs <-  ncol(gwaa@phdata[-c(1:2)])
ntests <- length(name)
Number_Sig_Hits=length(which(log10pvps>-1*log10(0.05/nsnps)))
Number_Sig_Hits_stringent <- length(which(log10pvps>-1*log10(0.05/
ntests)))
stringent_cpgs <- name[which(log10pvps>-1*log10(0.05/ntests))]
stringent_log_pvals <- log10pvps[which(log10pvps>-1*log10(0.05/
ntests))]
Highest_pval=max(log10pvps, na.rm=T)
window_name <- as.character(leisel$snp[i])
hits_leisel_detailed <- data.frame(window_name, nsnps, nCPGs,
ntests, Number_Sig_Hits, Number_Sig_Hits_stringent, Highest_pval,
stringent_cpgs, stringent_log_pvals)
write.csv(hits_leisel_detailed, file=paste("/Users/ecazaly/Desktop/
PhD_Analysis/Association_2015april/Ass_genetic_input/
Genetic_riskLoci/hits_leisel/Detail_hits/Detail_hits_",
as.character(leisel$snp[i]),".csv", collapse=""))
}
# this gives 53 files, the amount with stringent sig hits
# bind these in one csv
filenames <- list.files(path="/Users/ecazaly/Desktop/PhD_Analysis/
Association_2015april/Ass_genetic_input/Genetic_riskLoci/
hits_leisel/Detail_hits/", full.names=T)
library(plyr)
import.list <- llply(filenames, read.csv, header=T)
Detail_hits_all_leisel <- matrix(nrow=200, ncol=10)
library(reshape)
Detail_hits_all_leisel <- merge_recurse(import.list)
head(Detail_hits_all_leisel)
dim(Detail_hits_all_leisel) # [1] 163  10
write.csv(Detail_hits_all_leisel, file="/Users/ecazaly/Desktop/
PhD_Analysis/Association_2015april/Ass_genetic_input/
Genetic_riskLoci/hits_leisel/Detail_hits/
Detail_hits_all_leisel.csv")

# A lot of repetition in the CpGs because of the overlapping sliding
window and this spreadsheet also lists the significant hits of each
SNP with a CpG so that the most significant one can be determined.
IN the methylation approach, only the ..
```

```
#####################################################################
#########################
## 2. Prostate Cancer risk regions identified in published studies
through linkage analysis  ##
#####################################################################
#########################

# A list of 32 prostate cancer risk linkage regions from published
familial studies was generated on OMIM (www.ncbi.nlm.nih.gov/omim).
These regions were already annotated to the hg19 genome build so it
was not necessary to perform conversion by SNPnexus.

linkage_region <- read.csv("/Users/ecazaly/Desktop/PhD_Analysis/
Association_2015april/Perform_Ass/PCrisk/Lit_risk/Familial/
Familial_lit_regions_pos.csv", header=T)
linkage_region$range <- (linkage_region$start -linkage_region
$end)*-1
write.csv(linkage_region, "/Users/ecazaly/Desktop/PhD_Analysis/
Association_2015april/Perform_Ass/PCrisk/Lit_risk/Familial/
Familial_lit_regions_pos2.csv")

# Create sliding windows for association analysis
linkage_region <- read.csv("/Users/ecazaly/Desktop/PhD_Analysis/
Association_2015april/Perform_Ass/PCrisk/Lit_risk/Familial/
Familial_lit_regions_pos2.csv", header=T)
# Create data frame with the new windows:  50KB total
linkage_windows_50KB <- data.frame(linkage_region$Region_Identifier,
linkage_region$hg19_chr, linkage_region$range, linkage_region
$start-10000, linkage_region$end+10000)
colnames(linkage_windows_50KB) <- c("window_name", "chr",
"ROI_range", "window_start", "window_end")
linkage_windows_50KB$window_range <- (linkage_windows_50KB
$window_start - linkage_windows_50KB$window_end)*-1
# need to adjust so this is 50Kb for smaller windows
write.csv(linkage_windows_50KB, file="/Users/ecazaly/Desktop/
PhD_Analysis/Association_2015april/Ass_genetic_input/
linkage_windows_50KB")


# do the first and second window manually then the thirs onwards in
a loop.  For the largest region of 44000000 bp there should be
44000000/40000 = 1100 windows? it's not too worrying that there are
a lot as long as there are not too many significant hits, just need
to be aware of controlling FDR or keeping the cut off p-vals very
stringent

first_window_start <- vector()
first_window_end <- vector()
first_window_range <- vector()
for(i in 1:32){
first_window_start[i] <- if(linkage_region$range[i]< 30000)
linkage_region$start[i] - ((50000-linkage_region$range[i])/2) else
linkage_region$start[i]-10000
first_window_end[i] <- if(linkage_region$range[i]< 30000)
```

```
linkage_region$start[i] + ((50000-linkage_region$range[i])/2) +
linkage_region$range[i] else linkage_region$start[i]+40000 #then the
next window starts 10kb in of that end
first_window_range[i] <- first_window_end[i] -first_window_start[i]
}
first_window <- data.frame(linkage_region$Region_Identifier,
rep(1,32), linkage_region$range, linkage_region$hg19_chr,
first_window_start, first_window_end, first_window_range)
colnames(first_window) <- c("ROI", "windows_window", "ROI_range",
"chr", "windows_start", "windows_end", "windows_range")
# Keep creating these windows until all regions are broken down to
50KB windows, then rbind all the windows and sort by
Region_Identifier then window
second_window_start <- vector()
second_window_end <- vector()
second_window_range <- vector()
for(i in 1:32){
second_window_start[i] <- if(linkage_region$range[i]> 30000)
first_window_end[i]-10000 else NA
second_window_end[i] <- if(linkage_region$range[i]> 30000)
(first_window_end[i]+ 40000) else NA
second_window_range[i] <- if(is.na(second_window_end[i])) NA else
second_window_end[i] - second_window_start[i] }
second_window <- data.frame(linkage_region$Region_Identifier,
rep(2,32), linkage_region$range, linkage_region$hg19_chr,
second_window_start, second_window_end, second_window_range)
colnames(second_window) <- c("ROI", "windows_window", "ROI_range",
"chr", "windows_start", "windows_end", "windows_range")

# how do you know when to stop the windows? for the 3rd if original
ROI >60 ? that covers 2 x30 unique windows plus overlap?  just keep
adding 30 on each time as that's the unique window part?
third_window_start <- vector()
third_window_end <- vector()
third_window_range <- vector()
for(i in 1:32){
third_window_start[i] <- if(!(is.na(second_window_range[i])) &
linkage_region$range[i]> 60000) second_window_end[i]-10000 else NA
third_window_end[i] <- if(linkage_region$range[i]> 60000)
(second_window_end[i]+ 40000) else NA
third_window_range[i] <- if(is.na(third_window_end[i])) NA else
third_window_end[i] - third_window_start[i]
}
third_window <- data.frame(linkage_region$Region_Identifier,
rep(3,32), linkage_region$range, linkage_region$hg19_chr,
third_window_start, third_window_end, third_window_range)
colnames(third_window) <-c("ROI", "windows_window", "ROI_range",
"chr", "windows_start", "windows_end", "windows_range")

fourth_window_start <- vector()
fourth_window_end <- vector()
fourth_window_range <- vector()
for(i in 1:32){
fourth_window_start[i] <- if(!(is.na(third_window_range[i])) &
```

```
linkage_region$range[i]> 90000) third_window_end[i]-10000 else NA
fourth_window_end[i] <- if(linkage_region$range[i]> 90000)
(third_window_end[i]+ 40000) else NA
fourth_window_range[i] <- if(is.na(fourth_window_end[i])) NA else
fourth_window_end[i] - fourth_window_start[i]
}
fourth_window <- data.frame(linkage_region$Region_Identifier,
rep(4,32), linkage_region$range, linkage_region$hg19_chr,
fourth_window_start, fourth_window_end, fourth_window_range)
colnames(fourth_window) <- c("ROI", "windows_window", "ROI_range",
"chr", "windows_start", "windows_end", "windows_range")
# write the first few windows in a csv, check, then combine first
two with others
oneTo4 <- rbind(first_window, second_window, third_window,
fourth_window)
oneTo4 <- oneTo4[with(oneTo4, order(oneTo4$ROI,
oneTo4$windows_window)),]
write.csv(oneTo4, file="/Users/ecazaly/Desktop/PhD_Analysis/
Association_2015april/Ass_genetic_input/windows/oneTo4.csv")

# loop the remaining windows (start from the 3rd window, can then
doubke check the third and fourth against the manual windows from
above)
linkage_region$Region_Identifier <- as.character(linkage_region
$Region_Identifier)
linkage_region$Region_Identifier[1] <- gsub("1", "01",
linkage_region$Region_Identifier[1], fixed=T)
linkage_region$Region_Identifier[2] <- gsub("2", "02",
linkage_region$Region_Identifier[2], fixed=T)
linkage_region$Region_Identifier[3] <- gsub("3", "03",
linkage_region$Region_Identifier[3], fixed=T)
linkage_region$Region_Identifier[4] <- gsub("4", "04",
linkage_region$Region_Identifier[4], fixed=T)
linkage_region$Region_Identifier[5] <- gsub("5", "05",
linkage_region$Region_Identifier[5], fixed=T)
linkage_region$Region_Identifier[6] <- gsub("6", "06",
linkage_region$Region_Identifier[6], fixed=T)
linkage_region$Region_Identifier[7] <- gsub("7", "07",
linkage_region$Region_Identifier[7], fixed=T)
linkage_region$Region_Identifier[8] <- gsub("8", "08",
linkage_region$Region_Identifier[8], fixed=T)
linkage_region$Region_Identifier[9] <- gsub("9", "09",
linkage_region$Region_Identifier[9], fixed=T)
windows_start <- vector()
windows_end <- vector()
windows_range <- vector()
windows_window <- vector()
ROI <- vector()
ROI_range <- vector()
chr <- vector()
for(i in 1:32){
for(j in 1:35200){
windows_start[j] <- if(linkage_region$range[i]> 30000*j)
linkage_region$start[i]+30000+40000*j else NA
```

```
windows_end[j] <- if(linkage_region$range[i]> 30000*j)
(linkage_region$start[i]+30000+40000*j)+50000 else NA
windows_end[j] <- if(!(is.na(windows_end[j])) & windows_start[j] <
linkage_region$end[i]) windows_end[j] else NA
windows_range[j] <- if(is.na(windows_end[j])) NA else windows_end[j]
- windows_start[j]
windows_window[j] <- j+02 # j+2 reflects that the loop starts at the
third window
ROI[j] <- rep(linkage_region
$Region_Identifier[i],length(windows_window[j]))
chr[j] <- rep(linkage_region$hg19_chr[i],length(windows_window[j]))
ROI_range[j] <- rep(linkage_region
$range[i],length(windows_window[j]))
}
windows <- data.frame(ROI, windows_window, ROI_range, chr,
windows_start, windows_end, windows_range)
windows <- windows[1:(length(windows_range)-
length(which(is.na(windows_range)))),]
dim(windows)
write.csv(windows, file=(paste("/Users/ecazaly/Desktop/PhD_Analysis/
Association_2015april/Ass_genetic_input/windows/", "windows_ROI_",i,
".csv", sep="")))
}
# This works. Then combine the oneTo4 and order by ROI,
windows_window, check then remove then manual 3&4 windows
# Needed to change window.1 etc to just 1,2,3... because when it
came to ordering had the issue of 01 vs 1 in alphanumeric ordering

filenames <- list.files(path="/Users/ecazaly/Desktop/PhD_Analysis/
Association_2015april/Ass_genetic_input/windows/", full.names=T)
library(plyr)
import.list <- llply(filenames, read.csv, header=T)
library(reshape)
windows_all <- merge_recurse(import.list)
dim(windows_all) # 5797    8
head(windows_all)
windows_all[1:20,1:6]
# order by ROI, window
windows_all <- windows_all[with(windows_all, order(windows_all$ROI,
windows_all$windows_window)),]
# remove duplicate windows
windows_all$X <- paste(windows_all$ROI, "_", windows_all
$windows_window, sep="")
dim(windows_all)-dim(windows_all[unique(windows_all$X),]) # should
take out 2x32=64 but only 52 because .. 64-52=12 /2 = 6... 5 only
had one window?
rownames(windows_all) <- 1:length(windows_all$X)
windows_all <- windows_all[which(!(duplicated(windows_all$X))),]
dim(windows_all) #5745  8  (5797-52)
# remove NAs
length(windows_all$windows_start) - length(which(is.na(windows_all
$windows_start))) #5723
# there are 22 rows with NAs
windows_all <- windows_all[which(!(is.na(windows_all
```

```
$windows_start))),]
dim(windows_all)
colnames(windows_all)[1] <- "ROI_window"
write.csv(windows_all[,-c(2,3)], file="/Users/ecazaly/Desktop/
PhD_Analysis/Association_2015april/Ass_genetic_input/windows/
windows_all.csv")
windows_all2 <- read.csv("/Users/ecazaly/Desktop/PhD_Analysis/
Association_2015april/Ass_genetic_input/windows/windows_all.csv")
head(windows_all2)

# save the workspace, script and plink files and move over to Hydra
for faster computation
load("/home/ecazaly/Data/Aug2015_onwards/Ass_genetic_input/
Genomic_sites.RData")

# perform association in parallel
library(minfi)
library(GenABEL)
library(foreach)
library(doParallel)
library(parallel)
numCores <- 35  # detectCores()
cl <- makeCluster(numCores)
registerDoParallel(cl)

cpgs_genomic <- vector()
foreach(i=1:length(windows_all2$ROI_window)) %dopar% { #
system(paste(c("/opt/apps/plink-1.07-x86_64/plink --noweb --bfile /
home/ecazaly/Data/Aug2015_onwards/Ass_genetic_input/cutoff_15_final
--keep /home/ecazaly/Data/Aug2015_onwards/Ass_genetic_input/
keep_samples.txt --chr ", windows_all2$chr[i]," --from-bp ",
windows_all2$windows_start[i]," --to-bp ",
windows_all2$windows_end[i]," --recode --transpose --out /home/
ecazaly/Data/Aug2015_onwards/Ass_genetic_input/windows/",
as.character(windows_all2$ROI_window[i])),collapse=""))
# Create .raw file
convert.snp.tped(tpedfile=paste("/home/ecazaly/Data/Aug2015_onwards/
Ass_genetic_input/windows/",
as.character(windows_all2$ROI_window[i]),".tped", sep=""),
tfamfile=paste("/home/ecazaly/Data/Aug2015_onwards/
Ass_genetic_input/windows/",
as.character(windows_all2$ROI_window[i]),".tfam", sep=""),
outfile=paste("/home/ecazaly/Data/Aug2015_onwards/Ass_genetic_input/
windows/", as.character(windows_all2$ROI_window[i]),".raw", sep=""))
system(paste(c("/opt/apps/plink-1.07-x86_64/plink --noweb --bfile /
home/ecazaly/Data/Aug2015_onwards/Ass_genetic_input/cutoff_15_final
--keep /home/ecazaly/Data/Aug2015_onwards/Ass_genetic_input/
keep_samples.txt --chr ", windows_all2$chr[i]," --from-bp ",
windows_all2$windows_start[i]," --to-bp ",
windows_all2$windows_end[i]," --recode --out /home/ecazaly/Data/
Aug2015_onwards/Ass_genetic_input/windows/",
as.character(windows_all2$ROI_window[i])),collapse=""))

cpgs_genomic <-
```

```
rownames(Annotated_meth[which(Annotated_chr==windows_all2$chr[i] &
Annotated_meth$pos <windows_all2$windows_end[i] & Annotated_meth$pos
>windows_all2$windows_start[i]),])
        # if(length(cpgs_genomic)<=1) { next  # no cpgs in the
window
                # }
        cpgs_genomic <- cpgs_genomic[which(cpgs_genomic %in%
rownames(Meth_B))]
        Meth_B_genomic <- data.frame(Meth_B[cpgs_genomic,])
        Meth_B_genomic <- if(length(cpgs_genomic)==1)
t(Meth_B_genomic) else Meth_B_genomic
        colnames(Meth_B_genomic) <- topVariable_samples
        rownames(Meth_B_genomic) <- cpgs_genomic
        keep_genomic <-
data.frame(Meth_B_genomic[,which(colnames(Meth_B_genomic) %in%
keep_sampleNames$V2)])
        keep_genomic <- if(length(cpgs_genomic)==1) t(keep_genomic)
else keep_genomic
        rownames(keep_genomic) <- if(length(cpgs_genomic)==1)
cpgs_genomic else rownames(keep_genomic)
        dim(keep_genomic) # 1 39ID <-
matrix(colnames(keep_genomic))
        ID <- matrix(colnames(keep_genomic))
colnames(ID) <- "id"
female <- keep_sampleNames[keep_sampleNames$V5==2,]
female$V2
sex <- matrix(c(rep(1,17),0,1,1,0,0,0,rep(1,11),0,1,0,0,1))
colnames(sex) <- "sex"
pheno <- cbind(ID, sex, t(keep_genomic))
write.table(pheno, file=paste("/home/ecazaly/Data/Aug2015_onwards/
Ass_genetic_input/windows/Phenos/
pheno_",as.character(windows_all2$ROI_window[i]), ".txt", sep=""),
quote=FALSE)
}

gwaa=load.gwaa.data(phenofile=paste("/home/ecazaly/Data/
Aug2015_onwards/Ass_genetic_input/windows/Phenos/
pheno_",windows_all2$ROI_window[i], ".txt", sep=""),
genofile=paste("/home/ecazaly/Data/Aug2015_onwards/
Ass_genetic_input/windows/",
as.character(windows_all2$ROI_window[i]),".raw", sep=""),
force=TRUE)
gt=as.data.frame(as.numeric(gtdata(gwaa)))
gwaa@gtdata@gtps=gt
gwaa@phdata$id=gsub("_","-",gwaa@phdata$id)
rownames(gwaa@phdata)=gsub("_", "-", rownames(gwaa@phdata))
gwaa@gtdata@idnames=gsub("_", "-", gwaa@gtdata@idnames)
rownames(gwaa@gtdata@gtps)=gsub("_", "-",
rownames(gwaa@gtdata@gtps))
name=vector()
coefs=vector()
pvals=vector()
log10pvps=vector()
Number_Sig_Hits=vector()
```

```
Highest_pval=vector()
modelnum=1

for(cpg in colnames(gwaa@phdata[-c(1:2)])) {
  for(snp in colnames(gwaa@gtdata@gtps)){
        if(sum(gwaa@gtdata@gtps[snp], na.rm=T)<1) {
                next  }
        name[modelnum]= paste(cpg, snp, sep= "/")
    formula=as.formula(paste(gwaa@phdata[cpg], "~",
gwaa@gtdata@gtps[snp]))
    association=polygenic_hglm(formula, ibs_no, gwaa)
    coefs[modelnum]= summary(association$hglm)$FixCoefMat[2,1]
    pvals[modelnum]= summary(association$hglm)$FixCoefMat[2,4]
    log10pvps[modelnum]= -1*log10(pvals[modelnum])
    modelnum=modelnum+1
    }}
# plot, will not plot at this stage, just generate significant hit
data
# create .map file by not transposing
map_b <- read.table(paste("/home/ecazaly/Data/Aug2015_onwards/
Ass_genetic_input/windows/",
as.character(windows_all2$ROI_window[i]),".map",sep="",
collapse=""), as.is=T)
# cut down map to the snps that have pvals
# name_keep <- gsub(paste(colnames(gwaa@phdata[,-c(1:2)]), "/",
sep="")[i], "", name, fixed=TRUE)
# map_b <- map[c(which(name_keep %in% map$V2)),]
# should correct by /ntests not /nsnps but see if any are sig first
because may use Simes procedure rather than Bonferroni to correct if
some look possibly sig
nsnps  <- ncol(gwaa@gtdata@gtps)
nCPGs <-  ncol(gwaa@phdata[-c(1:2)])
ntests <- ncol(gwaa@gtdata@gtps) *  ncol(gwaa@phdata[-c(1:2)])
Number_Sig_Hits=length(which(log10pvps>-1*log10(0.05/nsnps)))
Number_Sig_Hits_stringent <- length(which(log10pvps>-1*log10(0.05/
ntests)))
Highest_pval=max(log10pvps, na.rm=T)
window_name <- windows_all2$ROI_window[i]
hits <- data.frame(window_name, nsnps, nCPGs, ntests,
Number_Sig_Hits, Number_Sig_Hits_stringent, Highest_pval)
write.csv(hits, file=paste("/home/ecazaly/Data/Aug2015_onwards/
Ass_genetic_input/hits/hits_",
as.character(windows_all2$ROI_window[i]),".csv", collapse=""))
  }

# Then bind together csvs
filenames <- list.files(path="/home/ecazaly/Data/Aug2015_onwards/
Ass_genetic_input/hits/hits_batch2", full.names=T)
library(plyr)
import.list <- llply(filenames, read.csv, header=T)
Hits_all_linkage <- matrix(nrow=2061, ncol=7)
library(reshape)
Hits_all_linkage_a <-
Hits_all_linkage_b <- merge_recurse(import.list)  # still too big
```

```
colnames(Hits_all_linkage) <- c("window_name", "nsnps", "nCPGs",
"ntests", "Number_Sig_Hits", "Number_Sig_Hits_stringent",
"Highest_pval")
write.csv(Hits_all_linkage, file="/home/ecazaly/Data/
Aug2015_onwards/Ass_genetic_input/hits2/Hits_all_linkage.csv")

# There are too many for the memory to handle.  Need to be more
stingent with the cut offs for significance, or the ones I choose to
bind.  Choose those with just log10 pvals >10 as for familial
associations
foreach(i=1:length(windows_all2$ROI_window)) %dopar% {
require(GenABEL)
gwaa=load.gwaa.data(phenofile=paste("/home/ecazaly/Data/
Aug2015_onwards/Ass_genetic_input/windows/Phenos/
pheno_",windows_all2$ROI_window[i], ".txt", sep=""),
genofile=paste("/home/ecazaly/Data/Aug2015_onwards/
Ass_genetic_input/windows/",
as.character(windows_all2$ROI_window[i]),".raw", sep=""),
force=TRUE)
gt=as.data.frame(as.numeric(gtdata(gwaa)))
gwaa@gtdata@gtps=gt
gwaa@phdata$id=gsub("_","-",gwaa@phdata$id)
rownames(gwaa@phdata)=gsub("_", "-", rownames(gwaa@phdata))
gwaa@gtdata@idnames=gsub("_", "-", gwaa@gtdata@idnames)
rownames(gwaa@gtdata@gtps)=gsub("_", "-",
rownames(gwaa@gtdata@gtps))
name=vector()
coefs=vector()
pvals=vector()
log10pvps=vector()
Number_Sig_Hits=vector()
Highest_pval=vector()
modelnum=1
for(cpg in colnames(gwaa@phdata[-c(1:2)])) {
  for(snp in colnames(gwaa@gtdata@gtps)){
        if(sum(gwaa@gtdata@gtps[snp], na.rm=T)<1) {
                next  }
        name[modelnum]= paste(cpg, snp, sep= "/")
    formula=as.formula(paste(gwaa@phdata[cpg], "~",
gwaa@gtdata@gtps[snp]))
    association=polygenic_hglm(formula, ibs_no, gwaa)
    coefs[modelnum]= summary(association$hglm)$FixCoefMat[2,1]
    pvals[modelnum]= summary(association$hglm)$FixCoefMat[2,4]
    log10pvps[modelnum]= -1*log10(pvals[modelnum])
    modelnum=modelnum+1
    }}
nsnps  <- ncol(gwaa@gtdata@gtps)
nCPGs <-  ncol(gwaa@phdata[-c(1:2)])
ntests <- length(name)
Number_Sig_Hits=length(which(log10pvps>-1*log10(0.05/nsnps)))
Number_Sig_Hits_stringent <- length(which(log10pvps>-1*log10(0.05/
ntests)))
Highest_pval=max(log10pvps, na.rm=T)
window_name <- windows_all2$ROI_window[i]
```

```
stringent_cpgs <- name[which(log10pvps>-1*log10(0.05/ntests))]
stringent_log_pvals <- log10pvps[which(log10pvps>-1*log10(0.05/
ntests))]
Highest_pval=max(log10pvps, na.rm=T)
hits_linkage_detailed <- data.frame(window_name, nsnps, nCPGs,
ntests, Number_Sig_Hits, Number_Sig_Hits_stringent, Highest_pval,
stringent_cpgs, stringent_log_pvals)
if (log10pvps[which(log10pvps>-1*log10(0.05/ntests))] <= 10)
write.csv(hits_linkage_detailed, file=paste("/home/ecazaly/Data/
Aug2015_onwards/Ass_genetic_input/hits/Detailed_hits_",
as.character(windows_all2$ROI_window[i]),".csv", collapse="")) #
Create excel files for just the ones with log pvals >10
  }
# The code works but not the if line as it writes csvs for hits with
lower log pvals than 10. I've upped the CPUs to 35 from 25, should
only take a few hours

# there are 231
# merge
filenames <- list.files(path="/home/ecazaly/Data/Aug2015_onwards/
Ass_genetic_input/hits/detailed/", full.names=T)
library(plyr)
import.list <- llply(filenames, read.csv, header=T)
Detail_hits_all_linkage <- matrix(nrow=400, ncol=10)
library(reshape)
Detail_hits_all_linkage <- merge_recurse(import.list)
head(Detail_hits_all_linkage)
dim(Detail_hits_all_linkage) # 573  10  The proportion with
significant hits is a lot less than the other methods, therefore not
an issue with FDRs??
# order by pval
Detail_hits_all_linkage_ord <-
Detail_hits_all_linkage[with(Detail_hits_all_linkage,
order(Detail_hits_all_linkage$stringent_log_pvals, decreasing=T)),]
write.csv(Detail_hits_all_linkage_ord, file="/home/ecazaly/Data/
Aug2015_onwards/Ass_genetic_input/hits/detailed/
Detail_hits_all_linkage_ord.csv")
# only 37 are above log pval of 10, manageable

hits_linkage <- read.csv("/Users/ecazaly/Desktop/PhD_Analysis/
Association_2015april/Ass_genetic_input/Genetic_riskLoci/
hits_linkage/Detail_hits_all_linkage_ord.csv")
# which have stringent pval hits?
hits_linkage_cpgs <- gsub("/..+", "", hits_linkage$stringent_cpgs)
length(hits_linkage_cpgs) #573
length(unique(hits_linkage_cpgs))  # 228, most out of all three but
not too huge considering how many started with
hits_linkage_cpgs_unique <- hits_linkage_cpgs[which(!
(duplicated(hits_linkage_cpgs)))]
# check how many have CpG_rs sites
Annotated_meth_hits_linkage <-
Annotated_meth[hits_linkage_cpgs_unique,]
hits_linkage_cpgs_rs <- Annotated_meth_hits_linkage[(which(!
(is.na(Annotated_meth_hits_linkage$CpG_rs)))),]
```

```
dim(hits_linkage_cpgs_rs)
# only 16, so much smaller proportion than the other 2  (6/13,
30/131)
hits_linkage_cpgs_rs$CpG_rs
# are any of these the original rs picked up in previous studies?
can't look at for this as it's regions but can with other two.  None
in the other two.   but could be in LD, look further into the
locations/proximites
rownames(hits_linkage_cpgs_rs)
# what proportion of sig pvals and >10 log pvals were these?
# 16/228 for sig hits.
 # [1] "cg16490124" "cg03964373" "cg11251367"
 # [4] "cg00069771" "cg16675926" "cg23209941"
 # [7] "cg07134368" "cg01021334" "cg00382740"
# [10] "cg20267322" "cg03224005" "cg04910228"
# [13] "cg13081429" "cg03272499" "cg24412204"
# [16] "cg02262873"
# 8/228 had log pvals >10 these are probably the most interesting.
7/8 of these had CpG_rs
# "cg16490124"
# "cg03964373"
# "cg11251367"
# "cg00069771"
# "cg16675926"
# "cg23209941"
# "cg07134368"
# the one that doesn't had quite a high log pval of 19.942
"cg24361198"
Annotated_meth_hits_linkage["cg24361198",]
# no probe SNPs, OpenSea, type II, chr1:242002464, quite far at the
end of q arm of chr1
# check genes within 1500 bases, as 450k annotation doesn't give any
there shouldn't be anything
242002464-1500   # chr1: 242000964
242002464+1500   # chr1: 242003964
# UCSC tell you if that CpG site has previously been found to be
methylated, unmethylated, partially methylated
# out of 6 diff cell lines 5 were methylated and 1 was unmethylated
in HepG2 (liver carcinoma cell line)

# Overlap with top[100 and top500] 95%-Range variable
length(which(hits_linkage_cpgs_unique %in% cpgInfo_range
$cpgNames_range))  #3
rownames(Annotated_meth_hits_linkage[which(hits_linkage_cpgs_unique
%in% cpgInfo_range$cpgNames_range),])
# "cg16490124" "cg11251367" "cg03224005"
# these all have CpG_rs
rownames(Annotated_meth_hits_linkage[which(hits_linkage_cpgs_unique
%in% cpgInfo_range$cpgNames_range),]) %in%
rownames(hits_linkage_cpgs_rs)  #[1] TRUE TRUE TRUE

length(which(hits_linkage_cpgs_unique %in%
cpgInfo_range_500$cpgNames_range_500)) #5
rownames(Annotated_meth_hits_linkage[which(hits_linkage_cpgs_unique
```

```
%in% cpgInfo_range_500$cpgNames_range_500),])
#[1] "cg16490124" "cg11251367" "cg00069771" "cg23209941"
"cg03224005"
# these don't overlap with the gwas ones
rownames(Annotated_meth_hits_linkage[which(hits_linkage_cpgs_unique
%in% cpgInfo_range_500$cpgNames_range_500),]) %in%
rownames(hits_linkage_cpgs_rs)  #[1] TRUE TRUE TRUE TRUE TRUE
# not overly surprising since 90/100 range were CpG_rs.. what about
500 proportion?

# what about overlap with Standard Deviation
length(which(hits_linkage_cpgs_unique %in% rownames(Top100_SD_M)))
#3, the same ones
rownames(Annotated_meth_hits_linkage[which(hits_linkage_cpgs_unique
%in% rownames(Top100_SD_M)),])
[1] "cg16490124" "cg11251367" "cg03224005"
length(which(hits_linkage_cpgs_unique %in% rownames(Top500_SD_M)))
#4 same as above but missing one
rownames(Annotated_meth_hits_linkage[which(hits_linkage_cpgs_unique
%in% rownames(Top500_SD_M)),])
[1] "cg16490124" "cg11251367" "cg00069771" "cg03224005"
# save all the proportions to /Users/ecazaly/Desktop/PhD_Analysis/
Association_2015april/Perform_Ass/proportions.xlsx

# Follow up the 8 CpGs with >10 log pvals, ie. check what genes
they're near and for the 7 that are CpG_rs if that snp has been
linked to disease risk etc
linkage_FollowUp_cpgs <- c("cg16490124", "cg03964373", "cg11251367",
"cg00069771", "cg16675926", "cg23209941", "cg07134368",
"cg24361198")
linkage_FollowUp <- Annotated_meth[linkage_FollowUp_cpgs,
c(1,2,4,9,12,14,15,16,17,18,19,24,28,29,33)]
# All on chr1.  They are all from ROI_8  230700000-243700000 Look
into this region/past studies more
write.csv(linkage_FollowUp, file="/Users/ecazaly/Desktop/
PhD_Analysis/Association_2015april/Ass_genetic_input/
Genetic_riskLoci/Hits_combined/linkage_FollowUp.csv")

# check any overlap with previously annotated variability by SD for
CpG_rs
linkage_FollowUp_cpgs  %in% rownames(top100sbeSNP) # two are
linkage_FollowUp_cpgs[which(linkage_FollowUp_cpgs  %in%
rownames(top100sbeSNP))]
[1] "cg16490124" "cg11251367"
# excel doc with gene info found at:
/Users/ecazaly/Desktop/PhD_Analysis/Association_2015april/
Perform_Ass/Top100variable/CPG_detail_long.xlsx

 save(linkage_FollowUp_cpgs, linkage_FollowUp, hits_linkage,
Annotated_meth_hits_linkage, hits_linkage_cpgs_rs, cpgInfo_range,
cpgInfo_range_500, ibs_no, Meth_B, Annotated_meth, Annotated_chr,
keep_sampleNames, topVariable_samples, file="/Users/ecazaly/Desktop/
PhD_Analysis/Association_2015april/Ass_genetic_input/
Genomic_PCrisk_linkage.RData")
```

```
#########################################################################
########
## 3. Prostate Cancer risk loci identified through published GWAS
studies ##
#########################################################################
########

# GWAS published PC risk loci generated from GWAS Catalog, then
generated windows of 25KB either side of risk SNP to create a 50KB
window of interest to perform association

# A list of 320 prostate cancer risk SNPs from published GWAS
studies was generated using the GWAS catalogue (https://
www.ebi.ac.uk/gwas/).  63 duplicate SNPs from multiple studies were
removed, along with 1 SNP with no information on UCSC, leaving 256
unique SNPs.

gwas_catalog <- read.delim("/Users/ecazaly/Desktop/PhD_Analysis/
Association_2015april/Perform_Ass/PCrisk/Lit_risk/gwas_risk/
GWAS_catalog_PROSTATE.tsv", header=T)
head(gwas_catalog)
dim(gwas_catalog)  # 320 variants, 33 data info columns

# column 22 "SNPs" has the rs numbers
# column 13 has the position as per genome build h38

gwas_catalog_lessINFO <- gwas_catalog[,c(11,12,13,22)]
gwas_catalog_lessINFO$db <- gsub("rs", "dbsnp  rs",
gwas_catalog_lessINFO$SNPs, fixed=T)
gwas_catalog_lessINFO$hg19_pos <-
c(rep("TBA",nrow(gwas_catalog_lessINFO)))
write.csv(gwas_catalog_lessINFO, file="/Users/ecazaly/Desktop/
PhD_Analysis/Association_2015april/Perform_Ass/PCrisk/Lit_risk/
gwas_risk/gwas_catalog_lessINFO.csv")
# when I put this into SNP nexus only got 254 back under "Genomic
Coordinates and External Links" and
# 494 under "Consequences on UCSC"

# Checked which were unique rs names:
length(unique(gwas_catalog_lessINFO$SNPs))  #[1] 257, still 3 more
than turned up in SNP nexus..
# but #25 is NA so that takes it down to 256
length(gwas_catalog_lessINFO$SNPs)  # [1] 320
# So there are 63 SNPs that double up in some studies?
length(gwas_catalog_lessINFO$SNPs) -
length(unique(gwas_catalog_lessINFO$SNPs)) #63
length(which(duplicated(gwas_catalog_lessINFO$SNPs))) #63
which(duplicated(gwas_catalog_lessINFO$SNPs))
unique_gwas <- gwas_catalog_lessINFO[-
c(which(duplicated(gwas_catalog_lessINFO$SNPs))),]
```

```
dim(unique_gwas) #257 6
length(unique(unique_gwas$SNPs)) #257
duplicated(unique_gwas$SNPs) # all FALSE
# remove row 23 as no info
unique_gwas <- unique_gwas[-23,]
write.csv(unique_gwas, file="/Users/ecazaly/Desktop/PhD_Analysis/
Association_2015april/Perform_Ass/PCrisk/Lit_risk/gwas_risk/
gwas_catalog_unique.csv")

# As these SNPs have been annotated to the latest genome build hg38,
I ran them through SNP nexus, to generate hg19 annotations so that
they can be compared to 450K data and other other PC risk info.
# Again got 254 out so went through ths excel file to find the two
missing ones, they are rs115457135 and rs11530697
# rs115457135 has now been 'merged' with rs7767188, replaced on
excel spreadsheet and looked up position
#  rs11530697 has now been 'merged' with rs3129859 , replaced on
excel spreadsheet and looked up position
# read back in and create window
unique_gwas2 <- read.csv("/Users/ecazaly/Desktop/PhD_Analysis/
Association_2015april/Perform_Ass/PCrisk/Lit_risk/gwas_risk/
gwas_catalog_unique.csv", head=T)
unique_gwas2$chromPosition
unique_gwas2$window <- paste(unique_gwas2$chromPosition-50000, ":",
unique_gwas2$chromPosition+50000, sep="")
write.csv(unique_gwas2, file="/Users/ecazaly/Desktop/PhD_Analysis/
Association_2015april/Perform_Ass/PCrisk/Lit_risk/gwas_risk/
gwas_catalog_unique2.csv")
# added to All_PCrisk_info spreadsheet


# generate windows for analysis
gwas <- read.csv("/Users/ecazaly/Desktop/PhD_Analysis/
Association_2015april/Perform_Ass/PCrisk/Lit_risk/gwas_risk/
gwas_catalog_unique2.csv")
dim(gwas) # 256 11
gwas <- gwas[,c(8,9,10)]
colnames(gwas) <- c("snp", "chr", "pos")
gwas$start_50KB <- gwas$pos-25000
gwas$end_50KB <- gwas$pos+25000
gwas$end_50KB - gwas$start_50KB # check
gwas$chr <- as.numeric(gsub("chr", "", gwas$chr, fixed=T))
# remove those on the X chr, I have no meth data for these
length(which(!(is.na(gwas$chr)))) #242
length(which(is.na(gwas$chr))) #14
gwas <- gwas[which(!(is.na(gwas$chr))),]
dim(gwas) # 242 5
gwas$snp

# Create .tped files
library(GenABEL)
for(snp in 1:length(gwas$snp)){
system(paste(c("/Applications/plink-1.07-mac-intel/plink --noweb --
bfile /Users/ecazaly/Desktop/PhD_Analysis/Association_2015april/
```

```
Ass_genetic_input/cutoff_15_final --keep /Users/ecazaly/Desktop/
PhD_Analysis/Association_2015april/Perform_Ass/Top100variable/
keep_samples.txt --chr ", gwas$chr[snp]," --from-bp ", gwas
$start_50KB[snp]," --to-bp ", gwas$end_50KB[snp]," --recode --
transpose --out /Users/ecazaly/Desktop/PhD_Analysis/
Association_2015april/Ass_genetic_input/Genetic_riskLoci/
windows_gwas/", as.character(gwas$snp[snp])),collapse=""))
convert.snp.tped(tpedfile=paste("/Users/ecazaly/Desktop/
PhD_Analysis/Association_2015april/Ass_genetic_input/
Genetic_riskLoci/windows_gwas/", as.character(gwas
$snp[snp]),".tped", sep=""), tfamfile=paste("/Users/ecazaly/Desktop/
PhD_Analysis/Association_2015april/Ass_genetic_input/
Genetic_riskLoci/windows_gwas/",as.character(gwas$snp[snp]),".tfam",
sep=""), outfile=paste("/Users/ecazaly/Desktop/PhD_Analysis/
Association_2015april/Ass_genetic_input/Genetic_riskLoci/
windows_gwas/", as.character(gwas$snp[snp]),".raw", sep=""))
# create .map file by not transposing
system(paste(c("/Applications/plink-1.07-mac-intel/plink --noweb --
bfile /Users/ecazaly/Desktop/PhD_Analysis/Association_2015april/
Ass_genetic_input/cutoff_15_final --keep /Users/ecazaly/Desktop/
PhD_Analysis/Association_2015april/Ass_genetic_input/
keep_samples.txt --chr ", gwas$chr[i]," --from-bp ", gwas
$start_50KB[i]," --to-bp ", gwas$end_50KB[i]," --recode --out /
Users/ecazaly/Desktop/PhD_Analysis/Association_2015april/
Ass_genetic_input/Genetic_riskLoci/windows_gwas/", as.character(gwas
$snp[i])),collapse=""))
}

# make pheno file
cpgs_genomic <- rownames(Annotated_meth[which(Annotated_chr==gwas
$chr[i] & Annotated_meth$pos <gwas$end_50KB[i] & Annotated_meth$pos
>gwas$start_50KB[i]),])
# if(length(cpgs_genomic)<=1) { next  # no cpgs in the window
# }
cpgs_genomic <- cpgs_genomic[which(cpgs_genomic %in%
rownames(Meth_B))]
Meth_B_genomic <- data.frame(Meth_B[cpgs_genomic,])
Meth_B_genomic <- if(length(cpgs_genomic)==1) t(Meth_B_genomic) else
Meth_B_genomic
colnames(Meth_B_genomic) <- topVariable_samples
rownames(Meth_B_genomic) <- cpgs_genomic
keep_genomic <-
data.frame(Meth_B_genomic[,which(colnames(Meth_B_genomic) %in%
keep_sampleNames$V2)])
keep_genomic <- if(length(cpgs_genomic)==1) t(keep_genomic) else
keep_genomic
rownames(keep_genomic) <- if(length(cpgs_genomic)==1) cpgs_genomic
else rownames(keep_genomic)
dim(keep_genomic) # 1 39
ID <- matrix(colnames(keep_genomic))
colnames(ID) <- "id"
female <- keep_sampleNames[keep_sampleNames$V5==2,]
female$V2
sex <- matrix(c(rep(1,17),0,1,1,0,0,0,rep(1,11),0,1,0,0,1))
```

```
colnames(sex) <- "sex"
dim(keep_genomic) # 39 1
pheno <- cbind(ID, sex, t(keep_genomic))
colnames(pheno)
write.table(pheno, file=paste("/Users/ecazaly/Desktop/PhD_Analysis/
Association_2015april/Ass_genetic_input/Genetic_riskLoci/
windows_gwas/pheno_",as.character(gwas$snp[i]), ".txt", sep=""),
quote=FALSE)

# Perform association
foreach(i=1:length(gwas$snp)) %dopar% {
        require(GenABEL)
gwaa=load.gwaa.data(phenofile=paste("/Users/ecazaly/Desktop/
PhD_Analysis/Association_2015april/Ass_genetic_input/
Genetic_riskLoci/windows_gwas/pheno_",as.character(gwas$snp[i]),
".txt", sep=""), genofile=paste("/Users/ecazaly/Desktop/
PhD_Analysis/Association_2015april/Ass_genetic_input/
Genetic_riskLoci/windows_gwas/", as.character(gwas$snp[i]),".raw",
sep=""),  force=TRUE)
gt=as.data.frame(as.numeric(gtdata(gwaa)))
gwaa@gtdata@gtps=gt
gwaa@phdata$id=gsub("_","-",gwaa@phdata$id)
rownames(gwaa@phdata)=gsub("_", "-", rownames(gwaa@phdata))
gwaa@gtdata@idnames=gsub("_", "-", gwaa@gtdata@idnames)
rownames(gwaa@gtdata@gtps)=gsub("_", "-",
rownames(gwaa@gtdata@gtps))
name=vector()
coefs=vector()
pvals=vector()
log10pvps=vector()
Number_Sig_Hits=vector()
Highest_pval=vector()
modelnum=1
for(cpg in colnames(gwaa@phdata[-c(1,2)])) {
  for(snp in colnames(gwaa@gtdata@gtps)){
        if(sum(gwaa@gtdata@gtps[snp], na.rm=T)<1) {
                next  }
        name[modelnum]= paste(cpg, snp, sep= "/")
    formula=as.formula(paste(gwaa@phdata[cpg], "~",
gwaa@gtdata@gtps[snp]))
    association=polygenic_hglm(formula, ibs_no, gwaa)
    coefs[modelnum]= summary(association$hglm)$FixCoefMat[2,1]
    pvals[modelnum]= summary(association$hglm)$FixCoefMat[2,4]
    log10pvps[modelnum]= -1*log10(pvals[modelnum])
    modelnum=modelnum+1
    }}
nsnps  <- ncol(gwaa@gtdata@gtps)
nCPGs <-  ncol(gwaa@phdata[-c(1:2)])
ntests <- length(name)
Number_Sig_Hits=length(which(log10pvps>-1*log10(0.05/nsnps)))
Number_Sig_Hits_stringent <- length(which(log10pvps>-1*log10(0.05/
ntests)))
Highest_pval=max(log10pvps, na.rm=T)
window_name <- as.character(gwas$snp[i])
```

```
stringent_cpgs <- name[which(log10pvps>-1*log10(0.05/ntests))]
stringent_log_pvals <- log10pvps[which(log10pvps>-1*log10(0.05/
ntests))]
Highest_pval=max(log10pvps, na.rm=T)
window_name <- as.character(gwas$snp[i])
hits_gwas_detailed <- data.frame(window_name, nsnps, nCPGs, ntests,
Number_Sig_Hits, Number_Sig_Hits_stringent, Highest_pval,
stringent_cpgs, stringent_log_pvals)
write.csv(hits_gwas_detailed, file=paste("/Users/ecazaly/Desktop/
PhD_Analysis/Association_2015april/Ass_genetic_input/
Genetic_riskLoci/hits_gwas/Detail_gwas_hits_",  as.character(gwas
$snp[i]),".csv", collapse=""))
}
# there are 79
# bind these together in one csv
filenames <- list.files(path="/Users/ecazaly/Desktop/PhD_Analysis/
Association_2015april/Ass_genetic_input/Genetic_riskLoci/hits_gwas/
Detail_hits/", full.names=T)
library(plyr)
import.list <- llply(filenames, read.csv, header=T)
Detail_hits_all_gwas <- matrix(nrow=400, ncol=10)
library(reshape)
Detail_hits_all_gwas <- merge_recurse(import.list)
head(Detail_hits_all_gwas)
dim(Detail_hits_all_gwas) # 745  10
write.csv(Detail_hits_all_gwas, file="/Users/ecazaly/Desktop/
PhD_Analysis/Association_2015april/Ass_genetic_input/
Genetic_riskLoci/hits_gwas/Detail_hits/Detail_hits_all_gwas.csv")
# These associations have one CpG with multiple SNPs, pull all the
CpGs that have at least one association greater than log10 pval of
10

hits_gwas <- read.csv("/Users/ecazaly/Desktop/PhD_Analysis/
Association_2015april/Ass_genetic_input/Genetic_riskLoci/hits_gwas/
Detail_hits/Detail_hits_all_gwas.csv")
head(hits_gwas)
# Need hits_gwas$stringent_cpgs
hits_gwas_cpgs <- gsub("/..+", "", hits_gwas$stringent_cpgs)
length(hits_gwas_cpgs) #745
length(unique(hits_gwas_cpgs))  # 131, so 131 separate CpGs
hits_gwas_cpgs_unique <- hits_gwas_cpgs[which(!
(duplicated(hits_gwas_cpgs)))]
# ones with log pval >10
hits_gwas_10plus <- hits_gwas[hits_gwas$stringent_log_pvals >=10,]
dim(hits_gwas_10plus) #155
hits_gwas_cpgs_10plus <- gsub("/..+", "", hits_gwas_10plus
$stringent_cpgs)
length(unique(hits_gwas_cpgs_10plus))  # 36, only 36 separate CpGs
with a log10 pval >10
 # [1] "cg11123619" "cg04741880" "cg26075039" "cg20720056"
"cg06221963"
 # [6] "cg09359103" "cg23069046" "cg02956194" "cg03036702"
"cg12474444"
# [11] "cg24634471" "cg04035553" "cg10596483" "cg14773235"
```

```
"cg18468917"
# [16] "cg16060930" "cg00366603" "cg10158182" "cg17239008"
"cg17351927"
# [21] "cg06281714" "cg06550200" "cg26690318" "cg14375985"
"cg09349613"
# [26] "cg13301327" "cg08564027" "cg19586845" "cg01452169"
"cg07146321"
# [31] "cg08241307" "cg06437931" "cg14271023" "cg14458575"
"cg16989719"
# [36] "cg13284789"

# which of these are GpG_rs
Annotated_meth_hits_gwas <- Annotated_meth[hits_gwas_cpgs_unique,]
hits_gwas_cpgs_rs <- Annotated_meth_hits_gwas[(which(!
(is.na(Annotated_meth_hits_gwas$CpG_rs)))),]
Annotated_meth_hits_gwas_10plus <-
Annotated_meth_hits_gwas[unique(hits_gwas_cpgs_10plus),]
dim(Annotated_meth_hits_gwas_10plus ) #36, 33
# which of the >10 logpvals have CpG_rs?
gwas_cpgs_rs_10plus <- Annotated_meth_hits_gwas_10plus[(which(!
(is.na(Annotated_meth_hits_gwas_10plus$CpG_rs)))),]
dim(gwas_cpgs_rs_10plus)
# 22 33, so 22 out of the 36 unique CpGs with a log10 >10 have a
CpG_rs
gwas_cpgs_rs_10plus$CpG_rs
 # [1] "rs75324250"  "rs45588133"  "rs112225149" "rs4919427"
"rs6920276"
 # [6] "rs672341"    "rs4714482"   "rs3888705"   "rs7485236"
"rs56209138"
# [11] "rs3134797"   "rs8192585"   "rs438475"    "rs28895028"
"rs12763379"
# [16] "rs72830824"  "rs72828989"  "rs11696871"  "rs12292212"
"rs57606101"
# [21] "rs35847523"  "rs4508746"

#  Follow up the 36 with >10pvals, ie. follow up 22 with a CpG_rs
and 14 that do not have CpG_rs
gwas_FollowUp_cpgs <- unique(hits_gwas_cpgs_10plus)
gwas_FollowUp <- Annotated_meth[gwas_FollowUp_cpgs,
c(1,2,4,9,12,14,15,16,17,18,19,24,28,29,33)]
write.csv(gwas_FollowUp, file="/Users/ecazaly/Desktop/PhD_Analysis/
Association_2015april/Ass_genetic_input/Genetic_riskLoci/
Hits_combined/gwas_FollowUp.csv")
# 6 have probe_rs sites and some are Type I probes with CpG_rs that
are not sbe_rs. Remove these as can't be sure the signal is not an
artifact

# Are any of these the original rs picked up in previous studies?
which(hits_gwas_cpgs_rs$CpG_rs %in% hits_gwas$window_name) # none

# what is the overlap with top[100 and top500] 95%-range variable
CpG sites?
length(which(hits_gwas_cpgs_unique %in% cpgInfo_range
$cpgNames_range))  #0
```

```
length(which(hits_gwas_cpgs_unique %in%
cpgInfo_range_500$cpgNames_range_500)) #6
rownames(Annotated_meth_hits_gwas[which(hits_gwas_cpgs_unique %in%
cpgInfo_range_500$cpgNames_range_500),])
[1] "cg02956194" "cg24634471" "cg10596483"
[4] "cg00366603" "cg26690318" "cg13301327"
# Overlap with Standard Deviation?
length(which(hits_gwas_cpgs_unique %in% rownames(Top100_SD_M))) #0
length(which(hits_gwas_cpgs_unique %in% rownames(Top500_SD_M))) #6
rownames(Annotated_meth_hits_gwas[which(hits_gwas_cpgs_unique %in%
rownames(Top500_SD_M)),])
[1] "cg02956194" "cg24634471" "cg10596483" "cg00366603"
[5] "cg26690318" "cg13301327"
# identical to 95%-range as above

# Create PNGs showing associations
# need to be specific with which cpg against all snps to get n(cpg)
plots
for(CPG in 1:ncol(gwaa@phdata[,-c(1:2)])){  #
png(filename=paste("/Users/ecazaly/Desktop/PhD_Analysis/
Association_2015april/Ass_genetic_input/Genetic_riskLoci/
PNGs_gwas/", colnames(gwaa@phdata[,-c(1:2)])[CPG],".png", sep=""),
pointsize=12,units="mm",width=137.6,height=137.6*2/3,res=800)
        par(family="serif")
        par(mar=c(4,4,3,0.5))
# I;m not sure how to subset the pvals so you just get the CPG ones
ie, /nsnps
        plot(map_b[,4]*10^-6,
log10pvps[],type="o",main=NULL,xlab=c("Position (Mb)",
cex=2),ylab="-1*log10(p-value)",xlim=c(map_b[1,4]*10^-6,map_b[nsnps,
4]*10^-6),ylim=c(-3,35),cex=0.2)
        title(main=list(paste("Association between ",
colnames(gwaa@phdata[,-c(1:2)])[i], " and ", nsnps, " SNPs in a 50Kb
window", sep=""), cex=.8))
        abline(h=-1*log10(0.05),lty=2)
        text(map_b[nsnps,4]*10^-6-0.005,-1*log10(0.05)+1,"p-value =
0.05",font=3, cex=0.4, col="red")
        abline(h=-1*log10(0.05/nsnps),lty=2)
        text(map_b[nsnps,4]*10^-6-0.005,-1*log10(0.05/nsnps)
+1.2,"adjusted p-value",font=3, cex=0.4, col="red")
        text(map_b[nsnps,4]*10^-6-0.033, 35, paste("Significant
associations: ", length(which(log10pvps[]> -1*log10(0.05/nsnps))),
sep=""), cex=0.6)
        # points(Annotated_meth[NC_info$cpgs[cpg],
2]*10^-6,-0.8,pch=17, col="red")
        # text(Annotated_meth[NC_info$cpgs[cpg],2]*10^-6,-3,
NC_info$cpgs[cpg], col="red", cex=0.8)
        dev.off()
        }
save(gwas, ibs_no, Meth_B, Annotated_meth, Annotated_chr,
keep_sampleNames, topVariable_samples, file="/Users/ecazaly/Desktop/
PhD_Analysis/Association_2015april/Ass_genetic_input/
Genomic_PCrisk_gwas.RData")
```

Apendix 5.1 Samples for which good quality bisulphite sequencing data was generated

	Sample ID	Resource Type	Sex	Disease Status	Age *
1	PC1-03	Familial	M	Affected	82
2	PC2-01	Familial	M	Affected	52
3	PC2-02	Familial	M	Affected	53
4	PC2-03	Familial	M	Affected	58
5	PC4-01	Familial	M	Affected	74
6	PC9-01	Familial	M	Affected	64
7	PC9-04	Familial	M	Affected	65
8	PC9-12	Familial	M	Affected	72
9	PC9-121	Familial	M	Unaffected	48
10	PC9-129	Familial	F	NA	61
11	PC9-24	Familial	F	NA	45
12	PC9-286	Familial	M	Unaffected	47
13	PC9-338	Familial	M	Affected	63
14	PC9-357	Familial	M	Unaffected	42
15	PC9-532	Familial	M	Affected	71
PC9 Median Age at Collection:					**62**
16	PC11-03	Familial	M	Affected	89
17	PC11-04	Familial	M	Affected	73
18	PC11-09	Familial	M	Affected	83
19	PC11-147	Familial	M	Affected	61
20	PC11-180	Familial	M	Unaffected	42
21	PC11-234	Familial	M	Unaffected	55
PC11 Median Age at Collection:					**67**
22	PC22-03	Familial	M	Affected	74
23	PC22-04	Familial	M	Affected	62
24	PC22-16	Familial	M	Affected	76
25	PC22-162	Familial	M	Unaffected	56
26	PC22-17	Familial	M	Affected	63
27	PC22-203	Familial	M	Affected	75
28	PC22-21	Familial	M	Affected	70
29	PC22-210	Familial	F	NA	73
30	PC22-274	Familial	M	Unaffected	45
31	PC22-387	Familial	M	Affected	79
32	PC22-388	Familial	M	Unaffected	73
33	PC22-393	Familial	F	NA	44

34	PC22-414	Familial	F	NA	66
35	PC22-416	Familial	M	Affected	61
36	PC22-418	Familial	M	Unaffected	54
37	PC22-468	Familial	M	Affected	69
38	PC22-476	Familial	M	Unaffected	36
PC22 Median Age at Collection:					**66**
39	PC27-01	Familial	M	Affected	64
40	PC72-01	Familial	M	Affected	71
41	PC72-02	Familial	M	Affected	85
42	PC72-03	Familial	M	Affected	70
43	PC72-04	Familial	M	Affected	78
44	PC72-106	Familial	M	Unaffected	46
45	PC72-126	Familial	M	Affected	49
46	PC72-187	Familial	F	NA	41
47	PC72-188	Familial	M	Other Cancer	23
48	PC72-77	Familial	M	Affected	75
PC72 Median Age at Collection:					**70**
49	PC75-01	Familial	M	Affected	65
50	DVA1690	Case/Control	M	Unaffected	70
51	DVA1694	Case/Control	M	Unaffected	68
52	DVA1695	Case/Control	M	Unaffected	70
53	DVA1696	Case/Control	M	Unaffected	68
54	DVA1703	Case/Control	M	Unaffected	71
55	DVA1704	Case/Control	M	Unaffected	68
56	DVA1705	Case/Control	M	Unaffected	70
57	DVA1710	Case/Control	M	Unaffected	67
58	DVA1711	Case/Control	M	Unaffected	69
59	DVA1718	Case/Control	M	Unaffected	67
60	DVA1720	Case/Control	M	Unaffected	69
61	DVA1723	Case/Control	M	Unaffected	67
62	DVA1725	Case/Control	M	Unaffected	70
63	DVA1728	Case/Control	M	Unaffected	67
64	DVA1729	Case/Control	M	Unaffected	68
65	DVA1730	Case/Control	M	Unaffected	70
66	DVA1732	Case/Control	M	Unaffected	68
67	DVA1735	Case/Control	M	Unaffected	69
68	DVA1737	Case/Control	M	Unaffected	69
69	DVA1739	Case/Control	M	Unaffected	70
70	DVA1743	Case/Control	M	Unaffected	68

71	DVA1744	Case/Control	M	Unaffected	69
72	DVA1745	Case/Control	M	Unaffected	70
73	DVA1746	Case/Control	M	Unaffected	67
74	DVA1747	Case/Control	M	Unaffected	68
75	DVA1748	Case/Control	M	Unaffected	70
76	DVA1749	Case/Control	M	Unaffected	71
77	DVA505	Case/Control	M	Unaffected	67
78	DVA552	Case/Control	M	Unaffected	67
79	DVA566	Case/Control	M	Unaffected	70
80	DVA576	Case/Control	M	Unaffected	68
81	DVA599	Case/Control	M	Unaffected	70
Control Median Age at Collection:					**69**

Median age of affected individuals: **71**
Median age of unaffected individuals: **69**

* Age at DNA collection in years
NA: non-applicable

Organge highlighting: 37 samples with good quality methylation and genotype data generated in Ch. 2-4

Appendix 5.2 Primers for bisulphite sequencing

	Gene	CpG	rs	chr	Position	Size
1	CASZ1_a	cg13387643	rs284310	1	10737562	930
2	CASZ1_b	cg13387643	rs284310	1	10737562	913
3	CDK2AP1	cg09084244	rs1109559	12	123757860	1102
4	MCC	cg08238375	rs4705795	5	112483149	1058
5	TP53INP2	cg20592836	rs2378256	20	33292126	1217
6	ITGB2	cg02464073	rs1721	21	46349496	1217
7	RPS6KA2	cg06330797	rs7357046	6	167195910	1230
8	USP7_a	cg01891583	rs2304466	16	8995926	817
9	USP7_b	cg01891583	rs2304466	16	8995926	822
10	S EPT9	cg05161773	rs426439	17	75378036	1183
11	MGMT	cg09993319	rs7898151	10	131529435	945
12	FOXP4	cg03036702	rs4714482	6	41528198	958
13	ARHGAP22	cg25013753	rs1051508	10	49654342	1109
14	C10orf46	cg00231519	rs36101953	10	120516119	946
15	AJAP1_a	cg00345083	rs7517857	1	4725584	735
16	AJAP1_b	cg00345083	rs7517857	1	4725584	864
17	NME6	cg08146865	rs3197223	3	48335857	751
18	ZFAT	cg21927991	rs5025124	8	135494242	1110
19	NSMCE4A	cg19360212	rs11200296	10	123731471	1296
20	FOXK2_a	cg05331763	rs79974293	17	80535367	905
21	FOXK2_b	cg05331763	rs79974293	17	80535367	880
22	FMN2	cg11251367	rs12403072	1	240620177	1180
23	TOX2	cg26365090	rs11700304	20	42574362	1201
24	SMC1B	cg17662493	rs6006744	22	45806309	1060
25	PRM1_a	cg02978201	rs737008	16	11374865	680
26	PRM1_b	cg02978201	rs737008	16	11374865	785
27	RAB11B_a	cg04610028	rs2967607	19	8464538	782
28	RAB11B_b	cg04610028	rs2967607	19	8464538	865
29	STK25	cg09289202	rs6757649	2	242443982	1000

Primer L	Primer R
TTTTTTTTATTTTGGTGTTTGTTTT	ATAACTATCCCCTATCCCAAAATAC
GTGGGTAAATAGGGAAGTAGTTGTT	ACTACTCAAACCCTCCACAAAAC
ATGTTTATTTAGGTTTGTTTTTTTT	ATTTCAAACAATTCTCCTATCTCAAC
TTATAGATTTTTAGTTTGAAAGGAAAAGAA	AAATATATCAAACAAACTTATAAACCC
TTTTTTTATATTTTTAGATGTTTTT	TAAAATCCCATACAACCCTAAACTC
TTTTATGGGGGTTTTTTAAGTTTAG	ACCCCCTACCTAACAAATACCTAAC
AAATTGGGAATTTTTTTGAGGATAG	ATAACCTCCATAATCACCCAAAAC
AATAGTTTAATGATAAGTGAAATGATAGTT	CTTAAACAATATTCCTAAAAACAATT
GGAAGAAGGTTTAATTTTGTGTGTT	ATTTCAAACAATTCTCCTACCTCAA
AGTTGGTTTTAAGTAGGGATTTTTG	TACCTCTCTCCCACCTAAATTATTT
TTTTTGTAGGTTTTTTTAAGTTTGTGTT	TAAACAACAATACCACTCTCCTCAA
TGGAAATTAGTTTGGGTAATAAAGTG	AAACCTAAAAACTCCTACAAATACC
AGAGTTAGGGTTATGGTGGAAAGTT	AAATCACAAACCCAAAAAAACTAAA
GTGTTTGTTGTTTTAGTTAGTTG	TCCAAAATAATCTTCTCATCTCAAAA
AAGTTAGTATTTTGTTGTGGATTTAATTTT	TCTTCCCAAAACATCAAACTTCTATA
ATTTTGGAGGGTTTTTTTGAATAT	CTACCCTACCCTCCTCCTAAATAAA
TTAAATGTTGTTTGGAGTTATTGTAT	AAAAATTACAACCTTCTTCCCTAAC
TTGGGTTATGGTATTATTTTGTGG	AAAAACATACTTCTCAAAAAAACTTCTC
TTAGGATTTTTTATTGGATTAAAAA	AAATAAAACAAACAAACACAAAATAC
TTGTTTTATTTTTATGGTGTTGTAATGT	ATAATCACAACAATTATCTCAATTCTAAAT
GGTGGAGGAGAAAGATAGTATAAAGTTT	TATTTCCTCACCCAACCCTAAATA
TAGATTTTTTTGATAGTGTTATGTT	CTACTCTATTTTTCCTTAAAACTTC
GGTTTTAGAAGAGTGGAAAGGAATTA	CTAACCCTACTCTCCAAAAAACAAC
TTTTTTGAGGAAGGTTTGGTTTA	CTTATCCATATACACTATCTTACAATAACC
TTAAGAGTTTGGAATAATGGTTAGG	TTCTTAATCTCACCAAAATACTACC
GTAGGTTTTTGATTTTTATTGGATG	AATAATACTTCTTAACAAAAACATATC
TGAGTAGTTGGGGTTATAGTAGTTATTA	AACAAACTCCTACCTTCCAAAAAC
TATTAGGTTGTAGGTGGAGGTTTAG	CAAATATAAACACCAATAAAAAAAA
TTTTTTTGTAGGTTTTTTTAAGTTTGTG	AACCCAACTATCTTCCACTAAAACA

Appendix 5.3 Optimal PCR conditions for meQTL regions

	Gene	CpG	Chr*	Position	Size	Temp**	Q solution	Cycles	Amplified
1	CASZ1_a	cg13387643	1	10737562	930	58	None	40	Yes
2	CASZ1_b	cg13387643	1	10737562	913	62	0.5uL	40	Yes
3	MCC	cg08238375	5	112483149	1058	56	None	40	Yes
4	S EPT9	cg05161773	17	75378036	1183	58	None	45	Yes
5	AJAP1_a	cg00345083	1	4725584	735	58	None	40	Yes
6	AJAP1_b	cg00345083	1	4725584	864	61	0.5uL	40	Yes
7	ZFAT	cg21927991	8	135494242	1110	61	0.5uL	45	Yes
8	SMC1B	cg17662493	22	45806309	1060	56	0.5uL	45	Yes
9	PRM1_a	cg02978201	16	11374865	680	58	None	40	Yes
10	PRM1_b	cg02978201	16	11374865	785	58	None	40	Yes
11	NME6	cg08146865	3	48335857	751	61	0.5uL	45	Yes
12	FOXP4	cg03036702	6	41528198	958	62	0.5uL	45	Yes
13	C10orf46	cg00231519	10	120516119	946	60	0.5uL	45	Yes
14	RPS6KA2	cg06330797	6	167195910	1230	60	0.5uL	45	Yes
15	MGMT	cg09993319	10	131529435	945	62	0.5uL	45	Yes
16	STK25	cg09289202		242443982	1000	62	0.5uL	45	Yes
17	ARHGAP22	cg25013753	10	49654342	1109	60	0.5uL	45	Yes
18	USP7_a	cg01891583	16	8995926	817	54	None	40	Yes
19	FOXK2_b	cg05331763	17	80535367	880	62	None	45	Yes
20	RAB11B_a	cg04610028	19	8464538	782	62	0.5uL	40	Yes
21	USP7_b	cg01891583	16	8995926	822	60	0.5uL	45	No
22	FOXK2_a	cg05331763	17	80535367	905	60	0.5uL	45	No
23	RAB11B_b	cg04610028	19	8464538	865	56	0.5uL	45	No
24	ITGB2	cg02464073	21	46349496	1217	60	0.5uL	45	No
25	NSMCE4A	cg19360212	10	123731471	1296	55	0.5uL	45	No
26	TP53INP2	cg20592836	20	33292126	1217	55	0.5uL	45	No
27	TOX2	cg26365090	20	42574362	1201	54	0.5uL	45	No
28	FMN2	cg11251367	1	240620177	1180	58	0.5uL	45	No
29	CDK2AP1	cg09084244	12	123757860	1102	60	0.5uL	45	No

* Chromosome

** Reaction Temperature

Quality Control Pipeline for Bisulphite Sequencing data:
Based on the Epigenesis guide http://www.epigenesys.eu/images/stories/protocols/pdf/20120720103700_p57.pdf

Key steps:
    # 1. QC
    # 2. Alignment
    # 3. Extract Methylation data
    # 4. Load Data in R
    # 5. Perform QC with BiSeq

1. Quality Control: FastQC (java): via command line or using a GUI
http://www.bioinformatics.babraham.ac.uk/projects/fastqc/
ssh ...
module avail FastQC
module load FastQC/11.4
cd /gd/apps/FastQC-11.4
./fastqc ~/Emma_home/BiSeq_wd/WD/*fastq.gz --o ~/Emma_home/BiSeq_wd/WD/FastQC_output

Quality and Adapter trimming: Trim Galore
This works on paired-end files, so both files per sample at the same time. Default: removes base calls with Phred score 20 or lower, removes Illumina adapter sequence form 3' end, removes sequences shorter than 20bp

cd ~/Emma_home/BiSeq_wd/trim_galore_zip
./trim_galore --paired --trim1 ~/Emma_home/BiSeq_wd/WD/*.fastq.gz --o ~/Emma_home/BiSeq_wd/test_2 --path_to_cutadapt ~/Emma_home/BiSeq_wd/cutadapt-1.9.1/bin/cutadapt

Generate a FastQC report for each trimmed read
cd /gd/apps/FastQC-11.4
./fastqc ~/Emma_home/BiSeq_wd/test_2/*.fq.gz --o ~/Emma_home/BiSeq_wd/FastQC_trimmed

2. Alignment to bisulphite reference: Bismark
Requires a working version of Perl, Bowtie2 and samtools
Sequencing data needs to be FastA format, either .fa or .fasta
download reference genome from Ensemble: http://

```
asia.ensembl.org/info/data/ftp/index.html/
# Genome build Hg19
# Path to UCSC reference genome:  /gd/Genome_Reference/gatk-
bundle/hg19/ucsc.hg19.fasta
# Copied over to my folder

cd /gd/apps/bowtie2-2.2.6
module load bowtie2
cd ~/Emma_home/BiSeq_wd/bismark_v0.15.0
chmod +x bismark
chmod +x bismark_genome_preparation

# Prepare the reference genome
./bismark_genome_preparation --path_to_bowtie /gd/apps/
bowtie2-2.2.6 --bowtie2 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome/

# Align the genome, use --multicore to process in parallel
./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/PC72-291_S90_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
PC72-291_S90_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/PC75-01_S74_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
PC75-01_S74_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/PC9-541_S33_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
PC9-541_S33_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/PC9-532_S75_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
PC9-532_S75_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
```

```
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/PC9-477_S2_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
PC9-477_S2_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/PC9-357_S62_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
PC9-357_S62_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/PC9-338_S14_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
PC9-338_S14_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/PC9-29_S4_L001_R1_001_val_1.fq.gz -2
~/Emma_home/BiSeq_wd/test_2/PC9-29_S4_L001_R2_001_val_2.fq.gz
--output_dir ~/Emma_home/BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/PC9-286_S26_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
PC9-286_S26_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/PC9-24_S78_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
PC9-24_S78_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
```

```
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/PC9-12_S83_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
PC9-12_S83_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/PC9-129_S86_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
PC9-129_S86_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/PC9-121_S36_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
PC9-121_S36_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/PC9-04_S32_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
PC9-04_S32_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/PC9-01_S22_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
PC9-01_S22_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/PC72-94_S84_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
PC72-94_S84_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/PC72-77_S29_L001_R1_001_val_1.fq.gz
```

```
-2 ~/Emma_home/BiSeq_wd/test_2/
PC72-77_S29_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/PC72-75_S9_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
PC72-75_S9_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/PC72-291_S90_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
PC72-291_S90_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/PC72-213_S73_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
PC72-213_S73_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/PC72-188_S16_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
PC72-188_S16_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/PC72-187_S8_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
PC72-187_S8_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/PC72-136_S52_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
PC72-136_S52_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
```

```
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/PC72-126_S79_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
PC72-126_S79_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/PC72-106_S3_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
PC72-106_S3_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/PC72-04_S25_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
PC72-04_S25_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/PC72-03_S5_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
PC72-03_S5_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/PC72-02_S94_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
PC72-02_S94_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/PC72-01_S50_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
PC72-01_S50_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output
```

```
./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/PC4-01_S7_L001_R1_001_val_1.fq.gz -2
~/Emma_home/BiSeq_wd/test_2/PC4-01_S7_L001_R2_001_val_2.fq.gz
--output_dir ~/Emma_home/BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/PC27-01_S82_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
PC27-01_S82_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/PC22-476_S54_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
PC22-476_S54_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/PC22-468_S34_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
PC22-468_S34_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/PC22-418_S6_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
PC22-418_S6_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/PC22-416_S48_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
PC22-416_S48_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/PC22-414_S55_L001_R1_001_val_1.fq.gz
```

```
-2 ~/Emma_home/BiSeq_wd/test_2/
PC22-414_S55_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/PC22-393_S91_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
PC22-393_S91_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/PC22-388_S31_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
PC22-388_S31_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/PC22-387_S93_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
PC22-387_S93_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/PC22-386_S41_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
PC22-386_S41_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/PC22-274_S87_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
PC22-274_S87_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/PC22-21_S89_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
PC22-21_S89_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
```

```
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/PC22-210_S64_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
PC22-210_S64_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/PC22-203_S30_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
PC22-203_S30_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/PC22-195_S19_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
PC22-195_S19_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/PC22-17_S95_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
PC22-17_S95_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/PC22-16_S72_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
PC22-16_S72_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/PC22-162_S28_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
PC22-162_S28_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output
```

```
./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/PC22-04_S47_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
PC22-04_S47_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/PC22-03_S53_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
PC22-03_S53_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/PC2-03_S18_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
PC2-03_S18_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/PC2-02_S80_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
PC2-02_S80_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/PC2-01_S20_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
PC2-01_S20_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/PC12-01_S61_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
PC12-01_S61_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
```

```
Emma_home/BiSeq_wd/test_2/PC11-415_S44_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
PC11-415_S44_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/PC11-234_S38_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
PC11-234_S38_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/PC11-233_S92_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
PC11-233_S92_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/PC11-180_S23_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
PC11-180_S23_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/PC11-147_S59_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
PC11-147_S59_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/PC11-09_S43_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
PC11-09_S43_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/PC11-04_S37_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
```

```
PC11-04_S37_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/PC11-03_S46_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
PC11-03_S46_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/PC1-03_S10_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
PC1-03_S10_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/NTC_S60_L001_R1_001_val_1.fq.gz -2
~/Emma_home/BiSeq_wd/test_2/NTC_S60_L001_R2_001_val_2.fq.gz --
output_dir ~/Emma_home/BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/DVA720_S77_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
DVA720_S77_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/DVA711_S76_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
DVA711_S76_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/DVA599_S42_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
DVA599_S42_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output
```

```
./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/DVA586_S96_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
DVA586_S96_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/DVA576_S85_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
DVA576_S85_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/DVA566_S67_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
DVA566_S67_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/DVA552_S24_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
DVA552_S24_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/DVA505_S45_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
DVA505_S45_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/DVA1749_S71_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
DVA1749_S71_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
```

```
Emma_home/BiSeq_wd/test_2/DVA1748_S1_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
DVA1748_S1_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/DVA1747_S27_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
DVA1747_S27_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/DVA1746_S11_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
DVA1746_S11_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/DVA1745_S56_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
DVA1745_S56_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/DVA1744_S68_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
DVA1744_S68_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/DVA1743_S40_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
DVA1743_S40_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/DVA1739_S17_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
```

```
DVA1739_S17_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/DVA1737_S15_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
DVA1737_S15_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/DVA1735_S51_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
DVA1735_S51_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/DVA1732_S66_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
DVA1732_S66_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/DVA1730_S58_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
DVA1730_S58_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/DVA1729_S63_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
DVA1729_S63_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/DVA1728_S70_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
DVA1728_S70_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output
```

```
./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/DVA1725_S65_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
DVA1725_S65_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/DVA1723_S13_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
DVA1723_S13_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/DVA1718_S57_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
DVA1718_S57_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/DVA1710_S88_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
DVA1710_S88_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/DVA1705_S39_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
DVA1705_S39_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/DVA1704_S81_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
DVA1704_S81_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
```

```
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/DVA1703_S21_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
DVA1703_S21_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/DVA1696_S35_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
DVA1696_S35_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/DVA1695_S12_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
DVA1695_S12_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/DVA1694_S49_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
DVA1694_S49_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

./bismark --multicore 10 -n 1 ~/Emma_home/BiSeq_wd/Alignment/
Ref_genome  --path_to_bowtie /gd/apps/bowtie2-2.2.6 -1 ~/
Emma_home/BiSeq_wd/test_2/DVA1690_S69_L001_R1_001_val_1.fq.gz
-2 ~/Emma_home/BiSeq_wd/test_2/
DVA1690_S69_L001_R2_001_val_2.fq.gz --output_dir ~/Emma_home/
BiSeq_wd/Alignment/alignment_output

# Creates two files (BAM and txt report file) for each of the
96 samples



3. Extract Methylation data
cd ~/Emma_home/BiSeq_wd/bismark_v0.15.0
chmod +x bismark_methylation_extractor  # makes this command
executable on the server
```

```
./bismark_methylation_extractor -s ~/Emma_home/BiSeq_wd/
Alignment/alignment_output/
DVA566_S67_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.bam -p --
no_overlap --o ~/Emma_home/BiSeq_wd/extracted_methylation_data
# makes 15 files per sample
# Do all
./bismark_methylation_extractor -s ~/Emma_home/BiSeq_wd/
Alignment/alignment_output/*.fq.gz_bismark_bt2_pe.bam -p --
no_overlap --o ~/Emma_home/BiSeq_wd/extracted_methylation_data
# Use the --comprehensive command so that it combines the 4
reads
./bismark_methylation_extractor -s --comprehensive ~/
Emma_home/BiSeq_wd/Alignment/alignment_output/
*.fq.gz_bismark_bt2_pe.bam -p --no_overlap --o ~/Emma_home/
BiSeq_wd/extracted_meth_comp

# Use bismark2bedGraph to get the required .cov files for R
chmod +x bismark2bedGraph
cd ~/Emma_home/BiSeq_wd/bismark_v0.15.0

./bismark2bedGraph --output DVA1748.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context/ ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_DVA1748_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output PC9-477.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_PC9-477_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output PC72-106.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_PC72-106_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.tx
t

./bismark2bedGraph --output PC9-29.bedGraph --dir ~/Emma_home/
BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/CpG_context/
CpG_context_PC9-29_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output PC72-03.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_PC72-03_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt
```

```
./bismark2bedGraph --output PC22-418.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_PC22-418_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.tx
t

./bismark2bedGraph --output PC4-01.bedGraph --dir ~/Emma_home/
BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/CpG_context/
CpG_context_PC4-01_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output PC72-187.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_PC72-187_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.tx
t

./bismark2bedGraph --output PC72-75.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_PC72-75_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output PC1-03.bedGraph --dir ~/Emma_home/
BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/CpG_context/
CpG_context_PC1-03_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output DVA1746.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_DVA1746_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output DVA1695.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_DVA1695_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output DVA1723.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_DVA1723_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output PC9-338.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
```

```
CpG_context_PC9-338_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output DVA1737.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_DVA1737_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output PC72-188.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_PC72-188_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.tx
t

./bismark2bedGraph --output DVA1739.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_DVA1739_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output PC2-03.bedGraph --dir ~/Emma_home/
BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/CpG_context/
CpG_context_PC2-03_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output PC22-195.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_PC22-195_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.tx
t

./bismark2bedGraph --output PC2-01.bedGraph --dir ~/Emma_home/
BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/CpG_context/
CpG_context_PC2-01_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output DVA1703.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_DVA1703_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output PC9-01.bedGraph --dir ~/Emma_home/
BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/CpG_context/
CpG_context_PC9-01_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output PC11-180.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
```

```
CpG_context_PC11-180_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.tx
t

./bismark2bedGraph --output DVA552.bedGraph --dir ~/Emma_home/
BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/CpG_context/
CpG_context_DVA552_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output PC72-04.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_PC72-04_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output PC9-286.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_PC9-286_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output DVA1747.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_DVA1747_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output PC22-162.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_PC22-162_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.tx
t

./bismark2bedGraph --output PC72-77.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_PC72-77_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output PC22-203.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_PC22-203_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.tx
t

./bismark2bedGraph --output PC22-388.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_PC22-388_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.tx
t
```

```
./bismark2bedGraph --output PC9-04.bedGraph --dir ~/Emma_home/
BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/CpG_context/
CpG_context_PC9-04_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output PC9-541.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_PC9-541_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output PC22-468.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_PC22-468_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.tx
t

./bismark2bedGraph --output DVA1696.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_DVA1696_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output PC9-121.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_PC9-121_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output PC11-04.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_PC11-04_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output PC11-234.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_PC11-234_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.tx
t

./bismark2bedGraph --output DVA1705.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_DVA1705_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output DVA1743.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
```

```
CpG_context/
CpG_context_DVA1743_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output PC22-386.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_PC22-386_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.tx
t

./bismark2bedGraph --output DVA599.bedGraph --dir ~/Emma_home/
BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/CpG_context/
CpG_context_DVA599_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output PC11-09.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_PC11-09_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output PC11-415.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_PC11-415_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.tx
t

./bismark2bedGraph --output DVA505.bedGraph --dir ~/Emma_home/
BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/CpG_context/
CpG_context_DVA505_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output PC11-03.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_PC11-03_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output PC22-04.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_PC22-04_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output PC22-416.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_PC22-416_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.tx
t
```

```
./bismark2bedGraph --output DVA1694.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_DVA1694_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output PC72-01.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_PC72-01_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output DVA1735.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_DVA1735_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output PC72-136.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_PC72-136_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.tx
t

./bismark2bedGraph --output PC22-03.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_PC22-03_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output PC22-476.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_PC22-476_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.tx
t

./bismark2bedGraph --output PC22-414.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_PC22-414_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.tx
t

./bismark2bedGraph --output DVA1745.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_DVA1745_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output DVA1718.bedGraph --dir ~/
```

```
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_DVA1718_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output DVA1730.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_DVA1730_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output PC11-147.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_PC11-147_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.tx
t

./bismark2bedGraph --output NTC.bedGraph --dir ~/Emma_home/
BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/CpG_context/
CpG_context_NTC_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output PC12-01.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_PC12-01_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output PC9-357.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_PC9-357_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output DVA1729.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_DVA1729_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output PC22-210.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_PC22-210_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.tx
t

./bismark2bedGraph --output DVA1725.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_DVA1725_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt
```

```
./bismark2bedGraph --output DVA1732.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_DVA1732_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output DVA566.bedGraph --dir ~/Emma_home/
BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/CpG_context/
CpG_context_DVA566_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output DVA1744.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_DVA1744_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output DVA1690.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_DVA1690_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output DVA1728.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_DVA1728_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output DVA1749.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_DVA1749_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output PC22-16.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_PC22-16_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output PC72-213.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_PC72-213_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.tx
t

./bismark2bedGraph --output PC75-01.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
```

```
CpG_context_PC75-01_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output PC9-532.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_PC9-532_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output DVA711.bedGraph --dir ~/Emma_home/
BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/CpG_context/
CpG_context_DVA711_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output DVA720.bedGraph --dir ~/Emma_home/
BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/CpG_context/
CpG_context_DVA720_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output PC9-24.bedGraph --dir ~/Emma_home/
BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/CpG_context/
CpG_context_PC9-24_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output PC72-126.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_PC72-126_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.tx
t

./bismark2bedGraph --output PC2-02.bedGraph --dir ~/Emma_home/
BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/CpG_context/
CpG_context_PC2-02_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output DVA1704.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_DVA1704_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output PC27-01.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_PC27-01_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output PC9-12.bedGraph --dir ~/Emma_home/
BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/CpG_context/
CpG_context_PC9-12_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output PC72-94.bedGraph --dir ~/
```

```
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_PC72-94_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output DVA576.bedGraph --dir ~/Emma_home/
BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/CpG_context/
CpG_context_DVA576_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output PC9-129.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_PC9-129_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output PC22-274.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_PC22-274_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.tx
t

./bismark2bedGraph --output DVA1710.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_DVA1710_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output PC22-21.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_PC22-21_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output PC72-291.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_PC72-291_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.tx
t

./bismark2bedGraph --output PC22-393.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_PC22-393_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.tx
t

./bismark2bedGraph --output PC11-233.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
```

```
CpG_context_PC11-233_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.tx
t

./bismark2bedGraph --output PC22-387.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_PC22-387_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.tx
t

./bismark2bedGraph --output PC72-02.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_PC72-02_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output PC22-17.bedGraph --dir ~/
Emma_home/BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/
CpG_context/
CpG_context_PC22-17_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt

./bismark2bedGraph --output DVA586.bedGraph --dir ~/Emma_home/
BiSeq_wd/R/CpG_context ~/Emma_home/BiSeq_wd/R/CpG_context/
CpG_context_DVA586_L001_R1_001_val_1.fq.gz_bismark_bt2_pe.txt



########################
## 4. Load Data in R ##
########################

library(BiSeq)
sample_sheet <- read.csv("SampleSheet_2.csv", header=T)
sample_names <- as.character(sample_sheet$Sample_ID)
# load all files
files <- list.files("~/Emma_home/BiSeq_wd/R/cov_files/")
file_path <- paste("~/Emma_home/BiSeq_wd/R/cov_files/", files,
sep="")
biseqRaw <- readBismark(file_path, colData=sample_names)
biseqRel <- rawToRel(biseqRaw) # create relative data



##############################
## 5. Perform QC with BiSeq ##
```

```
##############################

# Subset object to contain only regions of interest
# Currently 104,302 CpGs, many off target regions as final
 regions of interest only 678

regions <- read.csv("/Users/ecazaly/Desktop/Thesis/Results/
 Chapter_5/MiSeq/My_analysis/BiSeq_regions_r.csv")
# pull these out by row.name
biseq_regions <- biseqRaw[regions$row.name,]

# QC plots
covStats_regions <- covStatistics(biseq_regions)
covered_cpgs <- covStats_regions$Covered_CpG_sites
range(covered_cpgs)      #[1]  43 389             # before
 removing off-target 118, 14375
median(covered_cpgs)     #303  #old 1164
mean(covered_cpgs)       #[1] 289       # before removing off-
 target 1670
median <- covStats_regions$Median_coverage
range(median)            #[1] 2 1472         # before removing
 off-target 2 67
mean(median)             #[1] 390            # before removing
 off-target 11.5

covBoxplots(biseq_regions, col="cornflowerblue", las=2)
png(file="/Users/ecazaly/Desktop/Thesis/Results/Chapter_5/
 MiSeq/My_analysis/covBoxplots_regions.png", pointsize=12,
 units="mm", width=250, height=250*2/3, res=900)
covBoxplots(biseq_regions, col="cornflowerblue", las=2)
dev.off()

# Set QC thresholds for samples dependant on:
     # 1. Median_coverage: median of the coverage of the CpG
 sites covered per sample must be above 10

samples_100 <- biseq_regions[,which(covStats_regions
 $Covered_CpG_sites>=100)]
samples_100@colData$Sample_ID
covStats_samples_100 <- covStatistics(samples_100)
covStats_samples_100$Covered_CpG_sites

# now remove any that do not have a min coverage of 10
covStats_samples_100$Median_coverage
```

```
length(which(covStats_samples_100$Median_coverage>=10))  # 88
samples_100_10 <-
 samples_100[,which(covStats_samples_100$Median_coverage>=10)]
dim(samples_100_10) #678, 88
clean_raw_100 <- samples_100_10
clean_rel_100 <- rawToRel(clean_raw_100)

covBoxplots(clean_raw_100, col="cornflowerblue", las=2)  #
 looks slightly better?
png(file="/Users/ecazaly/Desktop/Thesis/Results/Chapter_5/
 MiSeq/My_analysis/clean100_covBoxplots_regions.png",
 pointsize=12, units="mm", width=250, height=250*2/3, res=900)
covBoxplots(clean_raw, col="cornflowerblue", las=2,
 main="Coverage per sample: clean data")
dev.off()

# Set QC thresholds for CpGs:
     # 1. Remove CpGs if row medians = 0
     # 2. Remove CpGs if row medians = 1
     # 3. Remove CpGs if NaN value is present in >90% of
 samples

length(which(rowMedians(methLevel(clean_rel_100),
 na.rm=T)==0)) #316
clean4 <- clean_raw_100[-
 which(rowMedians(methLevel(clean_rel_100), na.rm=T)==1),]
clean4_rel <- rawToRel(clean4)
clean5 <- clean4[-which(rowMedians(methLevel(clean4_rel),
 na.rm=T)==0),]
#253 sites, 88 samples
clean5_rel <- rawToRel(clean5)
length_nan_5 <- function(row) {
length(which(is.nan(methLevel(clean5_rel)[row,])))
}
number_nans_5 <- sapply(1:nrow(clean5_rel), length_nan_5)
length(which(number_nans_5>75.6))  # 5
clean_raw5_90nan <- clean5[-which(number_nans_5>75.6),]
dim(clean_raw5_90nan) #248, 88
clean_rel5_90nan <- rawToRel(clean_raw5_90nan)
```

Analysis Pipeline for Bisulphite Sequencing data in R

```
#####################################################################
#################################################
    # Key steps:
    #   1. Validate 450k data
    #   2. Examine genotype-methylation association
    #   3. Examine methylation levels between cancer/control groups
#####################################################################
#################################################


############################
## 1. Validate 450k data ##
############################

# This was performed on raw data not cleaned because I wanted to
include as many CpGs as possible when validating

# use the biseq_regions object generated in the above section

methLevel(bsRel_keyCpg)
length(methLevel(bsRel_keyCpg))  #1248 (96x13), this includes the
 NTC + 95 samples
# remove NTC
bsRel_keyCpg2 <- bsRel_keyCpg[,-34]
length(methLevel(bsRel_keyCpg2))  #1235 (95x13)
length(which(is.nan(methLevel(bsRel_keyCpg2)))) #199  16%

# write this data to a CSV and check against methylation array data
write.csv(t(methLevel(bsRel_keyCpg2)), file="/Users/ecazaly/Desktop/
 Thesis/Results/Chapter_5/MiSeq/My_analysis/
 450k_biseq_validation.csv")




################################################
## 2. Examine genotype-methylation association ##
################################################


# Methylation Landscape Plots in 37 samples from the familial study
with both methylation and genotype array data. These were the
samples analysed for meQTL analysis in Chapter.4 An additional 2
samples were removed from the analysis in this chapter as the
bisulphite sequencing data was of poor quality
```

```
# 1. Group samples into 3 groups based on genotype at the CpG of
 interest
# 2. Within each group take the median methylation at each CpG site.
# 3. Plot these three medians across all CpGs in the region using a
 different colour for each genotype
# 4. Is there a pattern between genotype and methylation?


# not present on Omni2.5 (8): FOXK2 (rs79974293), CASZ1 (rs284310),
 SEPT9 (rs426439), MGMT (rs7898151), MCC (rs4705795), FOXP4
 (rs4714482), C10orf46 (rs36101953), RAB11 (rs2967607)
# present on Omni2.5 (5): NME6 (rs3197223) , AJAP (rs7517857), USP7
 (kgp9608995), PRM1 (rs737008), ITGB2 (rs1721)



###  Regions with CpG-SNP present on Omni2.5  ###

# Plot with clean data:  clean_raw5_90nan
gwaa_all_omni@gtdata@idnames[4]=gsub("_a", "",
 gwaa_all_omni@gtdata@idnames[4])
gwaa_all_omni@gtdata@idnames[14]=gsub("_a", "",
 gwaa_all_omni@gtdata@idnames[14])
rownames(gwaa_all_omni@gtdata@gtps)[4]=gsub("_a", "",
 rownames(gwaa_all_omni@gtdata@gtps)[4])
rownames(gwaa_all_omni@gtdata@gtps)[14]=gsub("_a", "",
 rownames(gwaa_all_omni@gtdata@gtps)[14])
gwaa_all_omni@phdata$id=gsub("_", "-", gwaa_all_omni@phdata$id)
gwaa_all_omni@gtdata@idnames=gsub("_", "-",
 gwaa_all_omni@gtdata@idnames)
rownames(gwaa_all_omni@gtdata@gtps)=gsub("_", "-",
 rownames(gwaa_all_omni@gtdata@gtps))


NME6_detail <- regions[which(regions$Gene=="NME6"),]
NME6_raw <- clean_raw5_90nan[which(rownames(clean_raw5_90nan) %in%
 NME6_detail$row.name),]
NME6_rel <- rawToRel(NME6_raw)

**** have to remove the poor quality samples ****
rownames(ibs_no) %in% colnames(NME6_raw)  # 3 False
NME6_gwaa <- gwaa_all_omni[which(rownames(ibs_no) %in%
 colnames(NME6_raw)),which(colnames(gwaa_all_omni@gtdata)=="rs319722
 3")]
NME6_genotype <- as.genotype.gwaa.data(NME6_gwaa)
```

```
dim(NME6_genotype)

NME6_rel_samples <- NME6_rel[,which(colnames(NME6_rel) %in%
 rownames(NME6_genotype))]
NME6_rel_samples_ord <-
 NME6_rel_samples[,order(match(colnames(NME6_rel_samples),
 rownames(NME6_genotype)))]
colnames(NME6_rel_samples_ord)
identical(colnames(NME6_rel_samples_ord), rownames(NME6_genotype))
 #TRUE

# work out the median methylation level per genotype
NME6_genotype$sample <- rownames(NME6_genotype)
NME6_AA <- NME6_genotype$sample[which(NME6_genotype$rs3197223=="A/
 A")]
NME6_GA <- NME6_genotype$sample[which(NME6_genotype$rs3197223=="A/
 G")]
NME6_GG <- NME6_genotype$sample[which(NME6_genotype$rs3197223=="G/
 G")]
length(NME6_AA)  #10
length(NME6_GA)  #13, 1 less
length(NME6_GG)  #14, 1 less
# check this against general populaiton frequencies?

NME6_AA_meth <- methLevel(NME6_rel_samples_ord)[, NME6_AA]
dim(NME6_AA_meth)
NME6_AA_medians <- rowMedians(NME6_AA_meth, na.rm=T)
NME6_GA_meth <- methLevel(NME6_rel_samples_ord)[, NME6_GA]
NME6_GA_medians <- rowMedians(NME6_GA_meth, na.rm=T)
NME6_GG_meth <- methLevel(NME6_rel_samples_ord)[, NME6_GG]
NME6_GG_medians <- rowMedians(NME6_GG_meth, na.rm=T)

# are they different?  not really
NME6_AA_medians
NME6_GA_medians
NME6_GG_medians
# no longer NaNs but there are still some 1s/0s

library(gplots)
png(filename="/Users/ecazaly/Dropbox/Thesis/Chapter_5/Drafts/
 Genotype_methylation/NME6_landscape.png", pointsize=12, units="mm",
 width=200, height=250, res=400)
par(mfrow=c(2,1), mar=c(5,3.3,4,1)+0.1, family="serif")
plot(start(ranges(NME6_rel_samples_ord))*10^-6, NME6_AA_medians,
 type="o", pch=15, col="dark green", bg="dark green", xlab="",
 ylab="", main="Median Methylation by Genotype NME6
```

```
cg08146865 / rs3197223", ylim=c(0, 1.05))
title(ylab="Median Methylation", line=2.2, cex.lab=1.2)
title(xlab="Position (Mb)", line=2.2, cex.lab=1.2)
points(start(ranges(NME6_rel_samples_ord))*10^-6, NME6_GA_medians,
 type="b", pch=16, col="hotpink")
points(start(ranges(NME6_rel_samples_ord))*10^-6, NME6_GG_medians,
 type="b", pch=17, col="cornflowerblue")
abline(v=48335857*10^-6, col="black", lwd=1)
legend(48.335843,1.11, legend="CpG-SNP: UTR in open sea",
 col="black", bty="n", cex=0.8)
legend(x=48.3356, y=0.3, pch=17, col="cornflower blue", bty="n",
 legend="GG (14)")
legend(x=48.3356, y=0.2, pch=15, col="dark green", bty="n",
 legend="AA (10)")
legend(x=48.3356, y=0.1, pch=16, col="hotpink", bty="n", legend="GA
 (13)")
dev.off()


### AJAP (rs7517857) ###
AJAP_detail<- regions[which(regions$Gene=="AJAP1_a_b"),]
AJAP_raw <- clean_raw5_90nan[which(rownames(clean_raw5_90nan) %in%
 AJAP_detail$row.name),]
AJAP_rel <- rawToRel(AJAP_raw)
AJAP_gwaa <- gwaa_all_omni[which(rownames(ibs_no) %in%
 colnames(AJAP_raw)),which(colnames(gwaa_all_omni@gtdata)=="rs751785
 7")]
AJAP_genotype <- as.genotype.gwaa.data(AJAP_gwaa)

which(colnames(AJAP_rel) %in% rownames(AJAP_genotype))
AJAP_rel_samples <- AJAP_rel[,which(colnames(AJAP_rel) %in%
 rownames(AJAP_genotype))]
colnames(AJAP_rel_samples)
rownames(AJAP_genotype) # not the same
AJAP_rel_samples_ord <-
 AJAP_rel_samples[,order(match(colnames(AJAP_rel_samples),
 rownames(AJAP_genotype)))]
colnames(AJAP_rel_samples_ord)
identical(colnames(AJAP_rel_samples_ord), rownames(AJAP_genotype))
 #TRUE

# work out the median methylation level per genotype
AJAP_genotype$sample <- rownames(AJAP_genotype)
AJAP_AA <- AJAP_genotype$sample[which(AJAP_genotype$rs7517857 =="A/
 A")]
AJAP_GA <- AJAP_genotype$sample[which(AJAP_genotype$rs7517857 =="A/
 G")]
```

```
AJAP_GG <- AJAP_genotype$sample[which(AJAP_genotype$rs7517857 =="G/
 G")]
length(AJAP_AA)  #11 now 10
length(AJAP_GA)  #17 now 16
length(AJAP_GG)  #11 now 10

AJAP_AA_meth <- methLevel(AJAP_rel_samples_ord)[, AJAP_AA]
dim(AJAP_AA_meth)
AJAP_AA_medians <- rowMedians(AJAP_AA_meth, na.rm=T)
AJAP_GA_meth <- methLevel(AJAP_rel_samples_ord)[, AJAP_GA]
AJAP_GA_medians <- rowMedians(AJAP_GA_meth, na.rm=T)
AJAP_GG_meth <- methLevel(AJAP_rel_samples_ord)[, AJAP_GG]
AJAP_GG_medians <- rowMedians(AJAP_GG_meth, na.rm=T)
AJAP_AA_medians
AJAP_GA_medians
AJAP_GG_medians

png(filename="/Users/ecazaly/Dropbox/Thesis/Chapter_5/Drafts/
 Genotype_methylation/AJAP_landscape.png", pointsize=12, units="mm",
 width=200, height=250, res=400)
par(mfrow=c(2,1), mar=c(5,3.3,4,1)+0.1, family="serif")
plot(start(ranges(AJAP_rel_samples_ord))*10^-6, AJAP_AA_medians,
 type="o", pch=15, col="dark green", bg="green", xlab="", ylab="",
 main="Median Methylation by Genotype AJAP
cg00345083 / rs7517857")
title(ylab="Median Methylation", line=2.2, cex.lab=1.2)
title(xlab="Position (Mb)", line=2.2, cex.lab=1.2)
points(start(ranges(AJAP_rel_samples_ord))*10^-6, AJAP_GA_medians,
 type="o", pch=16, col="hotpink")
points(start(ranges(AJAP_rel_samples_ord))*10^-6, AJAP_GG_medians,
 type="o", pch=17, col="cornflowerblue")
abline(v=4725584*10^-6, col="black", lwd=1)
legend(4.72513,1.062, legend="CpG-SNP: intronic in CpG shore",
 col="black", bty="n", cex=0.8)
legend(x=4.72485, y=0.3, pch=17, col="cornflower blue", bty="n",
 legend="GG (10)")
legend(x=4.72485, y=0.2, pch=15, col="dark green", bty="n",
 legend="AA (10)")
legend(x=4.72485, y=0.1, pch=16, col="hotpink", bty="n", legend="GA
 (16)")
dev.off()


### USP7 (kgp9608995), cg01891583, 8995926 ###
USP7_detail <- regions[which(regions$Gene=="USP7_a"),]
USP7_raw <- clean_raw5_90nan[which(rownames(clean_raw5_90nan) %in%
 USP7_detail$row.name),]
```

```
USP7_rel <- rawToRel(USP7_raw)
USP7_gwaa <- gwaa_all_omni[which(rownames(ibs_no) %in%
 colnames(USP7_raw)),which(colnames(gwaa_all_omni@gtdata)=="kgp96089
 95")]
USP7_genotype <- as.genotype.gwaa.data(USP7_gwaa)
which(colnames(USP7_rel) %in% rownames(USP7_genotype))
USP7_rel_samples <- USP7_rel[,which(colnames(USP7_rel) %in%
 rownames(USP7_genotype))]
colnames(USP7_rel_samples)
rownames(USP7_genotype) # not the same
USP7_rel_samples_ord <-
 USP7_rel_samples[,order(match(colnames(USP7_rel_samples),
 rownames(USP7_genotype)))]
colnames(USP7_rel_samples_ord)
identical(colnames(USP7_rel_samples_ord), rownames(USP7_genotype))
 #TRUE
USP7_genotype$sample <- rownames(USP7_genotype)
USP7_AA <- USP7_genotype$sample[which(USP7_genotype$kgp9608995 =="A/
 A")]
USP7_GA <- USP7_genotype$sample[which(USP7_genotype$kgp9608995 =="A/
 G")]
USP7_GG <- USP7_genotype$sample[which(USP7_genotype$kgp9608995 =="G/
 G")]
length(USP7_AA)  #14 now 13
length(USP7_GA)  #20 now 19
length(USP7_GG)  #4
USP7_AA_meth <- methLevel(USP7_rel_samples_ord)[, USP7_AA]
dim(USP7_AA_meth)
USP7_AA_medians <- rowMedians(USP7_AA_meth, na.rm=T)
USP7_GA_meth <- methLevel(USP7_rel_samples_ord)[, USP7_GA]
USP7_GA_medians <- rowMedians(USP7_GA_meth, na.rm=T)
USP7_GG_meth <- methLevel(USP7_rel_samples_ord)[, USP7_GG]
USP7_GG_medians <- rowMedians(USP7_GG_meth, na.rm=T)
USP7_AA_medians
USP7_GA_medians
USP7_GG_medians

png(filename="/Users/ecazaly/Dropbox/Thesis/Chapter_5/Drafts/
 Genotype_methylation/USP7_landscape.png", pointsize=12, units="mm",
 width=200, height=250, res=400)
par(mfrow=c(2,1), mar=c(5,3.3,4,1)+0.1, family="serif")
plot(start(ranges(USP7_rel_samples_ord))*10^-6, USP7_AA_medians,
 type="o", pch=15, col="green", bg="green", xlab="", ylab="",
 main="Median Methylation by Genotype USP7
cg01891583 / kgp9608995", ylim=c(0, 1.06))
title(ylab="Median Methylation", line=2.2, cex.lab=1.2)
title(xlab="Position (Mb)", line=2.2, cex.lab=1.2)
```

```
points(start(ranges(USP7_rel_samples_ord))*10^-6, USP7_GA_medians,
 type="o", pch=16, col="hotpink")
points(start(ranges(USP7_rel_samples_ord))*10^-6, USP7_GG_medians,
 type="o", pch=17, col="cornflowerblue")
abline(v= 8995926*10^-6, col="black", lwd=1)
legend(8.99574,1.12, legend="CpG-SNP: intronic in open sea",
 col="black", bty="n", cex=0.8)
legend(x=8.99533, y=0.3, pch=17, col="cornflower blue", bty="n",
 legend="GG (14)")
legend(x=8.99533, y=0.2, pch=15, col="dark green", bty="n",
 legend="AA (10)")
legend(x=8.99533, y=0.1, pch=16, col="hotpink", bty="n", legend="GA
 (13)")
dev.off()


### PRM1 (rs737008), cg02978201, 11374865 ###
PRM1_detail <- regions[which(regions$Gene=="PRM1_a_b"),]
PRM1_raw <- clean_raw_90nan[which(rownames(clean_raw_90nan) %in%
 PRM1_detail$row.name),]
PRM1_rel <- rawToRel(PRM1_raw)
PRM1_gwaa <- gwaa_all_omni[which(rownames(ibs_no) %in%
 colnames(PRM1_raw)),which(colnames(gwaa_all_omni@gtdata)=="rs737008
 ")]
PRM1_genotype <- as.genotype.gwaa.data(PRM1_gwaa)
which(colnames(PRM1_rel) %in% rownames(PRM1_genotype))
PRM1_rel_samples <- PRM1_rel[,which(colnames(PRM1_rel) %in%
 rownames(PRM1_genotype))]
colnames(PRM1_rel_samples)
rownames(PRM1_genotype) # not the same
PRM1_rel_samples_ord <-
 PRM1_rel_samples[,order(match(colnames(PRM1_rel_samples),
 rownames(PRM1_genotype)))]
colnames(PRM1_rel_samples_ord)
identical(colnames(PRM1_rel_samples_ord), rownames(PRM1_genotype))
 #TRUE
PRM1_genotype$sample <- rownames(PRM1_genotype)
PRM1_AA <- PRM1_genotype$sample[which(PRM1_genotype$rs737008 =="A/
 A")]
PRM1_CA <- PRM1_genotype$sample[which(PRM1_genotype$rs737008 =="A/
 C")]
PRM1_CC <- PRM1_genotype$sample[which(PRM1_genotype$rs737008 =="C/
 C")]
length(PRM1_AA)  #20, 19 now
length(PRM1_CA)  #17, 15 now
length(PRM1_CC)  #2
PRM1_AA_meth <- methLevel(PRM1_rel_samples_ord)[, PRM1_AA]
```

```
dim(PRM1_AA_meth)
PRM1_AA_medians <- rowMedians(PRM1_AA_meth, na.rm=T)
PRM1_CA_meth <- methLevel(PRM1_rel_samples_ord)[, PRM1_CA]
PRM1_CA_medians <- rowMedians(PRM1_CA_meth, na.rm=T)
PRM1_CC_meth <- methLevel(PRM1_rel_samples_ord)[, PRM1_CC]
PRM1_CC_medians <- rowMedians(PRM1_CC_meth, na.rm=T)
PRM1_AA_medians
PRM1_CA_medians
PRM1_CC_medians

png(filename="/Users/ecazaly/Dropbox/Thesis/Chapter_5/Drafts/
 Genotype_methylation/PRM1_landscape.png", pointsize=12, units="mm",
 width=200, height=250, res=400)
par(mfrow=c(2,1), mar=c(5,3.3,4,1)+0.1, family="serif")
plot(start(ranges(PRM1_rel_samples_ord))*10^-6, PRM1_AA_medians,
 type="o", pch=15, col="dark green", bg="green", xlab="", ylab="",
 ylim=c(0, 1.06), main="Median Methylation by Genotype PRM1
cg02978201 / rs737008")
title(ylab="Median Methylation", line=2.2, cex.lab=1.2)
title(xlab="Position (Mb)", line=2.2, cex.lab=1.2)
points(start(ranges(PRM1_rel_samples_ord))*10^-6, PRM1_CA_medians,
 type="o", pch=16, col="hotpink")
points(start(ranges(PRM1_rel_samples_ord))*10^-6, PRM1_CC_medians,
 type="o", pch=17, col="cornflowerblue")
abline(v= 11374865*10^-6, col="black", lwd=1)
legend(11.37483,1.12, legend="CpG-SNP: coding (synonymous in open
 sea)", col="black", bty="n", cex=0.8)
legend(x=11.37425, y=0.3, pch=17, col="cornflower blue", bty="n",
 legend="CC (2)")
legend(x=11.37425, y=0.2, pch=15, col="dark green", bty="n",
 legend="AA (19)")
legend(x=11.37425, y=0.1, pch=16, col="hotpink", bty="n", legend="CA
 (15)")
dev.off()


### ITGB2 (rs1721), cg02464073 ###
ITGB2_detail <- regions[which(regions$Gene=="ITGB2"),]
ITGB2_raw <- clean_raw5_90nan[which(rownames(clean_raw5_90nan) %in%
 ITGB2_detail$row.name),]
ITGB2_rel <- rawToRel(ITGB2_raw)
ITGB2_gwaa <- gwaa_all_omni[which(rownames(ibs_no) %in%
 colnames(ITGB2_raw)),which(colnames(gwaa_all_omni@gtdata)=="rs1721"
 )]
ITGB2_genotype <- as.genotype.gwaa.data(ITGB2_gwaa)
which(colnames(ITGB2_rel) %in% rownames(ITGB2_genotype))
ITGB2_rel_samples <- ITGB2_rel[,which(colnames(ITGB2_rel) %in%
```

```
 rownames(ITGB2_genotype))]
colnames(ITGB2_rel_samples)
rownames(ITGB2_genotype) # not the same
ITGB2_rel_samples_ord <-
 ITGB2_rel_samples[,order(match(colnames(ITGB2_rel_samples),
 rownames(ITGB2_genotype)))]
colnames(ITGB2_rel_samples_ord)
identical(colnames(ITGB2_rel_samples_ord), rownames(ITGB2_genotype))
 #TRUE
ITGB2_genotype$sample <- rownames(ITGB2_genotype)
ITGB2_AA <- ITGB2_genotype$sample[which(ITGB2_genotype$rs1721 =="A/
 A")]
ITGB2_GA <- ITGB2_genotype$sample[which(ITGB2_genotype$rs1721 =="A/
 G")]
ITGB2_GG <- ITGB2_genotype$sample[which(ITGB2_genotype$rs1721 =="G/
 G")]
length(ITGB2_AA)  #2, now 1
length(ITGB2_GA)  #24, now 23
length(ITGB2_GG)  #12, now 11
ITGB2_AA_meth <- methLevel(ITGB2_rel_samples_ord)[, ITGB2_AA]
dim(ITGB2_AA_meth)
ITGB2_AA_medians <- ITGB2_AA_meth # only 1 change to meth value
 instead of rowMedians
ITGB2_GA_meth <- methLevel(ITGB2_rel_samples_ord)[, ITGB2_GA]
ITGB2_GA_medians <- rowMedians(ITGB2_GA_meth, na.rm=T)
ITGB2_GG_meth <- methLevel(ITGB2_rel_samples_ord)[, ITGB2_GG]
ITGB2_GG_medians <- rowMedians(ITGB2_GG_meth, na.rm=T)
ITGB2_AA_medians
ITGB2_GA_medians
ITGB2_GG_medians

png(filename="/Users/ecazaly/Dropbox/Thesis/Chapter_5/Drafts/
 Genotype_methylation/ITGB2_landscape.png", pointsize=12,
 units="mm", width=200, height=250, res=400)
par(mfrow=c(2,1), mar=c(5,3.3,4,1)+0.1, family="serif")
plot(start(ranges(ITGB2_rel_samples_ord))*10^-6, ITGB2_AA_medians,
 type="o", pch=15, col="dark green", bg="green", xlab="", ylab="",
 main="Median Methylation by Genotype ITGB2
cg02464073 / rs1721", ylim=c(0, 1.06))
title(ylab="Median Methylation", line=2.2, cex.lab=1.2)
title(xlab="Position (Mb)", line=2.2, cex.lab=1.2)
points(start(ranges(ITGB2_rel_samples_ord))*10^-6, type="o",
 ITGB2_GA_medians, pch=16, col="hotpink")
points(start(ranges(ITGB2_rel_samples_ord))*10^-6, type="o",
 ITGB2_GG_medians, pch=17, col="cornflowerblue")
abline(v= 46349496*10^-6, col="black", lwd=1)
legend(46.349475,1.12, legend="CpG-SNP: UTR in CpG shore",
```

```
 col="black", bty="n", cex=0.8)
legend(x=46.34929, y=0.3, pch=17, col="cornflower blue", bty="n",
 legend="GG (11)")
legend(x=46.34929, y=0.2, pch=15, col="dark green", bty="n",
 legend="AA (1)")
legend(x=46.34929, y=0.1, pch=16, col="hotpink", bty="n", legend="GA
 (23)")
dev.off()



###  Regions without CpG-SNP on Omni2.5  ###

# CASZ1
CASZ1_detail <- regions[which(regions$Gene=="CASZ1_a_b"),]
CASZ1_raw <- clean_raw5_90nan[which(rownames(clean_raw5_90nan) %in%
 CASZ1_detail$row.name),]
CASZ1_rel <- rawToRel(CASZ1_raw)
# genotype at kpg1150744   or rs284307
casz1_gwaa <-
 gwaa_all_omni[,which(colnames(gwaa_all_omni@gtdata)=="rs284307")]
casz1_genotype <- as.genotype.gwaa.data(casz1_gwaa)
# need to convert kpg1150744 to rs
# casz1_gwaa2 <-
 gwaa_all_omni[,which(colnames(gwaa_all_omni@gtdata)=="kpg1150744")]
# casz1_genotype2 <- as.genotype.gwaa.data(casz1_gwaa2)
casz1_genotype$sample <- rownames(casz1_genotype)
casz1_AA <- casz1_genotype$sample[which(casz1_genotype$rs284307=="A/
 A")]
casz1_GA <- casz1_genotype$sample[which(casz1_genotype$rs284307
 =="A/G")]
casz1_GG <- casz1_genotype$sample[which(casz1_genotype$rs284307
 =="G/G")]
length(casz1_AA)  #11
length(casz1_GA)  #15
length(casz1_GG)  #13

which(colnames(CASZ1_rel) %in% rownames(casz1_genotype))
CASZ1_rel_samples <- CASZ1_rel[,which(colnames(CASZ1_rel) %in%
 rownames(casz1_genotype))]
colnames(CASZ1_rel_samples)
rownames(casz1_genotype) # not the same
CASZ1_rel_samples_ord <-
 CASZ1_rel_samples[,order(match(colnames(CASZ1_rel_samples),
 rownames(casz1_genotype)))]
colnames(CASZ1_rel_samples_ord)
```

```
identical(colnames(CASZ1_rel_samples_ord), rownames(casz1_genotype))
 #FALSE, 2 geno samples not having seq data
which(!(rownames(casz1_genotype) %in%
 colnames(CASZ1_rel_samples_ord)))
# 16 (PC9-477), 23 (PC9-29)

casz1_geno <- casz1_genotype[which(casz1_genotype$sample %in%
 colnames(CASZ1_rel_samples_ord)),]
identical(colnames(CASZ1_rel_samples_ord), rownames(casz1_geno))
 #TRUE
casz1_AA <- casz1_geno$sample[which(casz1_geno$rs284307=="A/A")]
casz1_GA <- casz1_geno$sample[which(casz1_geno$rs284307 =="A/G")]
casz1_GG <- casz1_geno$sample[which(casz1_geno$rs284307 =="G/G")]
length(casz1_AA)  #9
length(casz1_GA)  #15
length(casz1_GG)  #13

CASZ1_AA_meth <- methLevel(CASZ1_rel_samples_ord)[, casz1_AA]
CASZ1_AA_medians <- rowMedians(CASZ1_AA_meth, na.rm=T)
CASZ1_GA_meth <- methLevel(CASZ1_rel_samples_ord)[, casz1_GA]
CASZ1_GA_medians <- rowMedians(CASZ1_GA_meth, na.rm=T)
CASZ1_GG_meth <- methLevel(CASZ1_rel_samples_ord)[, casz1_GG]
CASZ1_GG_medians <- rowMedians(CASZ1_GG_meth, na.rm=T)
library(gplots)
png(filename="/Users/ecazaly/Dropbox/Thesis/Chapter_5/Drafts/
 Genotype_methylation/CASZ1_landscape.png", pointsize=12,
 units="mm", width=200, height=250, res=400)
par(mfrow=c(2,1), mar=c(5,3.3,4,1)+0.1, family="serif")
plot(start(ranges(CASZ1_rel_samples_ord))*10^-6, CASZ1_AA_medians,
 type="b", pch=15, col="cornflower blue", bg="green", xlab="",
 ylab="", main="Median Methylation by Genotype CASZ1", ylim=c(0,
 1.06))
title(ylab="Median Methylation", line=2.2, cex.lab=1.2)
title(xlab="Position (Mb)", line=2.2, cex.lab=1.2)
points(start(ranges(CASZ1_rel_samples_ord))*10^-6, type="b",
 CASZ1_GA_medians, pch=16, col="dark green")
points(start(ranges(CASZ1_rel_samples_ord))*10^-6, type="b",
 CASZ1_GG_medians, pch=17, col="hotpink")
abline(v=10737562*10^-6, col="black", lwd=1)
legend(10.73731,1.12, legend="CpG-SNP: intronic in open sea",
 col="black", bty="n", cex=0.8)
legend(x=10.7369, y=0.3, pch=15, col="cornflower blue", bty="n",
 legend="AA (9)") #AA
legend(x=10.7369, y=0.2, pch=16, col="dark green", bty="n",
 legend="GA (15)") #GA
legend(x=10.7369, y=0.1, pch=17, col="hotpink", bty="n", legend="GG
 (13)") #GG
```

```
dev.off()

# C10orf46:   cg00231519 / rs36101953 / chr10:120516119

# rs10886296: 120522232 - 120516119  #6113
# rs4319431: 120515182 - 120516119     # -937
# Use rs4319431 genotype as this is ~1kb away and the LD plot
 indicates in LD with CpG-SNP
which(colnames(gwaa_all_omni@gtdata)=="rs4319431") # [1] 987672


C10orf46_gwaa <-
 gwaa_all_omni[,which(colnames(gwaa_all_omni@gtdata)=="rs4319431")]
C10orf46_genotype <- as.genotype.gwaa.data(C10orf46_gwaa)
C10orf46_genotype$sample <- rownames(C10orf46_genotype)
C10orf46_AA <- C10orf46_genotype$sample[which(C10orf46_genotype
 $rs4319431 =="A/A")]
C10orf46_GA <- C10orf46_genotype$sample[which(C10orf46_genotype
 $rs4319431 =="A/G")]
C10orf46_GG <- C10orf46_genotype$sample[which(C10orf46_genotype
 $rs4319431 =="G/G")]
length(C10orf46_AA)  #5
length(C10orf46_GA)  #10
length(C10orf46_GG)  #24
# there's 39 but should only be 37 as 2 biseq are bad quality.
 Perform below to eliminate and get geno data in right format

C10orf46_detail <- regions[which(regions$Gene=="C10orf46"),]
C10orf46_raw <- clean_raw5_90nan[which(rownames(clean_raw5_90nan)
 %in% C10orf46_detail$row.name),]
C10orf46_rel <- rawToRel(C10orf46_raw)

which(colnames(C10orf46_rel) %in% rownames(C10orf46_genotype))  #37
C10orf46_rel_samples <- C10orf46_rel[,which(colnames(C10orf46_rel)
 %in% rownames(C10orf46_genotype))]
colnames(C10orf46_rel_samples)
rownames(C10orf46_genotype) # not the same
C10orf46_rel_samples_ord <-
 C10orf46_rel_samples[,order(match(colnames(C10orf46_rel_samples),
 rownames(C10orf46_genotype)))]
colnames(C10orf46_rel_samples_ord)
identical(colnames(C10orf46_rel_samples_ord),
 rownames(C10orf46_genotype)) #FALSE, 2 geno samples not having seq
 data
which(!(rownames(C10orf46_genotype) %in%
 colnames(C10orf46_rel_samples_ord)))
# 16 (PC9-477), 23 (PC9-29)
```

```
C10orf46_geno <- C10orf46_genotype[which(C10orf46_genotype$sample
 %in% colnames(C10orf46_rel_samples_ord)),]
identical(colnames(C10orf46_rel_samples_ord),
 rownames(C10orf46_geno))  #TRUE
C10orf46_AA <- C10orf46_geno$sample[which(C10orf46_geno$rs4319431
 =="A/A")]
C10orf46_GA <- C10orf46_geno$sample[which(C10orf46_geno$rs4319431
 =="A/G")]
C10orf46_GG <- C10orf46_geno$sample[which(C10orf46_geno$rs4319431
 =="G/G")]
length(C10orf46_AA)  # 5
length(C10orf46_GA)  # 9
length(C10orf46_GG)  # 23

C10orf46_AA_meth <- methLevel(C10orf46_rel_samples_ord)[,
 C10orf46_AA]
C10orf46_AA_medians <- rowMedians(C10orf46_AA_meth, na.rm=T)
C10orf46_GA_meth <- methLevel(C10orf46_rel_samples_ord)[,
 C10orf46_GA]
C10orf46_GA_medians <- rowMedians(C10orf46_GA_meth, na.rm=T)
C10orf46_GG_meth <- methLevel(C10orf46_rel_samples_ord)[,
 C10orf46_GG]
C10orf46_GG_medians <- rowMedians(C10orf46_GG_meth, na.rm=T)

png(file="/Users/ecazaly/Dropbox/Thesis/Chapter_5/Drafts/
 Genotype_methylation/from_LD/C10orf46_landscape.png", pointsize=12,
 units="mm", width=200, height=250, res=400)
par(mfrow=c(2,1), mar=c(5,3.3,4,1)+0.1, family="serif")
plot(start(ranges(C10orf46_rel_samples_ord))*10^-6,
 C10orf46_AA_medians, type="b", pch=15, col="cornflower blue",
 bg="green", xlab="", ylab="", main="Median Methylation by Genotype
 C10orf46
  cg00231519 / rs36101953", ylim=c(0,1.07))
title(ylab="Median Methylation", line=2.2, cex.lab=1.2)
title(xlab="Position (Mb)", line=2.2, cex.lab=1.2)
points(start(ranges(C10orf46_rel_samples_ord))*10^-6, type="b",
 C10orf46_GA_medians, pch=16, col="dark green")
points(start(ranges(C10orf46_rel_samples_ord))*10^-6, type="b",
 C10orf46_GG_medians, pch=17, col="hotpink")
abline(v=120516119*10^-6, col="black", lwd=1)
legend(120.51587, 1.12, legend="CpG-SNP: intronic in CpG shore",
 col="black", bty="n", cex=0.8)
legend(x=120.5155, y=0.3, pch=15, col="cornflower blue", bty="n",
 legend="AA (5)") #AA
legend(x=120.5155, y=0.2, pch=16, col="dark green", bty="n",
 legend="GA (9)") #GA
```

```
legend(x=120.5155, y=0.1, pch=17, col="hotpink", bty="n", legend="GG
 (23)") #GG
dev.off()


# RAB11: cg04610028 / rs2967607 / chr19:8464538 / intronic in CpG
 shore

conversion[which(conversion$Name=="kgp6405269"),] #rs2913970
 8464653 - 8464538 = 115
conversion[which(conversion$Name=="kgp9275677"),] #rs2913971
 8463460 - 8464538 = -1078

# Use genotype at kgp6405269 as this is ~100b away and the LD plot
 indicates in LD with CpG-SNP
which(colnames(gwaa_all_omni@gtdata)=="kgp6405269") #[1] 1467904

RAB11_gwaa <-
 gwaa_all_omni[,which(colnames(gwaa_all_omni@gtdata)=="kgp6405269")]
RAB11_genotype <- as.genotype.gwaa.data(RAB11_gwaa)
RAB11_genotype$sample <- rownames(RAB11_genotype)
RAB11_AA <- RAB11_genotype$sample[which(RAB11_genotype
 $kgp6405269=="A/A")]
RAB11_GA <- RAB11_genotype$sample[which(RAB11_genotype
 $kgp6405269=="A/G")]
RAB11_GG <- RAB11_genotype$sample[which(RAB11_genotype
 $kgp6405269=="G/G")]
length(RAB11_AA)  # 2
length(RAB11_GA)  # 11
length(RAB11_GG)  # 26

RAB11_detail <- regions[which(regions$Gene=="RAB11_a"),]
RAB11_raw <- clean_raw5_90nan[which(rownames(clean_raw5_90nan) %in%
 RAB11_detail$row.name),]
RAB11_rel <- rawToRel(RAB11_raw)

which(colnames(RAB11_rel) %in% rownames(RAB11_genotype))  #37
RAB11_rel_samples <- RAB11_rel[,which(colnames(RAB11_rel) %in%
 rownames(RAB11_genotype))]
colnames(RAB11_rel_samples)
rownames(RAB11_genotype) # not the same
RAB11_rel_samples_ord <-
 RAB11_rel_samples[,order(match(colnames(RAB11_rel_samples),
 rownames(RAB11_genotype)))]
colnames(RAB11_rel_samples_ord)
identical(colnames(RAB11_rel_samples_ord), rownames(RAB11_genotype))
 #FALSE, 2 geno samples not having seq data
```

```
which(!(rownames(RAB11_genotype) %in%
 colnames(RAB11_rel_samples_ord)))
# 16 (PC9-477), 23 (PC9-29)


RAB11_geno <- RAB11_genotype[which(RAB11_genotype$sample %in%
 colnames(RAB11_rel_samples_ord)),]
identical(colnames(RAB11_rel_samples_ord), rownames(RAB11_geno))
 #TRUE
RAB11_AA <- RAB11_geno$sample[which(RAB11_geno$kgp6405269 =="A/A")]
RAB11_GA <- RAB11_geno$sample[which(RAB11_geno$kgp6405269 =="A/G")]
RAB11_GG <- RAB11_geno$sample[which(RAB11_geno$kgp6405269 =="G/G")]
length(RAB11_AA)  # 2
length(RAB11_GA)  # 10
length(RAB11_GG)  # 25

RAB11_AA_meth <- methLevel(RAB11_rel_samples_ord)[, RAB11_AA]
RAB11_AA_medians <- rowMedians(RAB11_AA_meth, na.rm=T)
RAB11_GA_meth <- methLevel(RAB11_rel_samples_ord)[, RAB11_GA]
RAB11_GA_medians <- rowMedians(RAB11_GA_meth, na.rm=T)
RAB11_GG_meth <- methLevel(RAB11_rel_samples_ord)[, RAB11_GG]
RAB11_GG_medians <- rowMedians(RAB11_GG_meth, na.rm=T)

png(file="/Users/ecazaly/Dropbox/Thesis/Chapter_5/Drafts/
 Genotype_methylation/from_LD/RAB11_landscape.png", pointsize=12,
 units="mm", width=200, height=250, res=400)
par(mfrow=c(2,1), mar=c(5,3.3,4,1)+0.1, family="serif")
plot(start(ranges(RAB11_rel_samples_ord))*10^-6, RAB11_AA_medians,
 type="b", pch=15, col="cornflower blue", bg="green", xlab="",
 ylab="", main="Median Methylation by Genotype RAB11
cg04610028 / rs2967607", ylim=c(0,1.07))
title(ylab="Median Methylation", line=2.2, cex.lab=1.2)
title(xlab="Position (Mb)", line=2.2, cex.lab=1.2)
points(start(ranges(RAB11_rel_samples_ord))*10^-6, type="b",
 RAB11_GA_medians, pch=16, col="dark green")
points(start(ranges(RAB11_rel_samples_ord))*10^-6, type="b",
 RAB11_GG_medians, pch=17, col="hotpink")
abline(v=8464538*10^-6, col="black", lwd=1)
legend(8.46432, 1.13, legend="CpG-SNP: intronic in CpG shore",
 col="black", bty="n", cex=0.8)
legend(x=8.463937, y=0.3, pch=15, col="cornflower blue", bty="n",
 legend="AA (2)") #AA
legend(x=8.463937, y=0.2, pch=16, col="dark green", bty="n",
 legend="GA (10)") #GA
legend(x=8.463937, y=0.1, pch=17, col="hotpink", bty="n", legend="GG
 (25)") #GG
dev.off()
```

```
# SEPT9: cg05161773 / rs426439 / chr17:75378036
conversion[which(conversion$Name=="kgp9468443"),]  #rs67129266
 75381103 - 75378036 = 3067
conversion[which(conversion$Name=="kgp9643238"),]  #rs312828
 75373938 - 75378036 = -4098

# Use genotype at kgp9468443 as this is ~3kb away and the LD plot
 indicates in LD with CpG-SNP
which(colnames(gwaa_all_omni@gtdata)=="")

SEPT9_gwaa <-
 gwaa_all_omni[,which(colnames(gwaa_all_omni@gtdata)=="kgp9468443")]
SEPT9_genotype <- as.genotype.gwaa.data(SEPT9_gwaa)
SEPT9_genotype$sample <- rownames(SEPT9_genotype)
SEPT9_AA <- SEPT9_genotype$sample[which(SEPT9_genotype$kgp9468443
 =="A/A")]
SEPT9_GA <- SEPT9_genotype$sample[which(SEPT9_genotype$kgp9468443
 =="A/G")]
SEPT9_GG <- SEPT9_genotype$sample[which(SEPT9_genotype$kgp9468443
 =="G/G")]
length(SEPT9_AA)  #17
length(SEPT9_GA)  #13
length(SEPT9_GG)  #9
# there's 39 but should only be 37 as 2 biseq are bad quality.
 Perform below to eliminate and get geno data in right format

SEPT9_detail <- regions[which(regions$Gene=="SEPT_9"),]
SEPT9_raw <- clean_raw5_90nan[which(rownames(clean_raw5_90nan) %in%
 SEPT9_detail$row.name),]
SEPT9_rel <- rawToRel(SEPT9_raw)

which(colnames(SEPT9_rel) %in% rownames(SEPT9_genotype))  #37
SEPT9_rel_samples <- SEPT9_rel[,which(colnames(SEPT9_rel) %in%
 rownames(SEPT9_genotype))]
colnames(SEPT9_rel_samples)
rownames(SEPT9_genotype) # not the same
SEPT9_rel_samples_ord <-
 SEPT9_rel_samples[,order(match(colnames(SEPT9_rel_samples),
 rownames(SEPT9_genotype)))]
colnames(SEPT9_rel_samples_ord)
identical(colnames(SEPT9_rel_samples_ord), rownames(SEPT9_genotype))
 #FALSE, 2 geno samples not having seq data
which(!(rownames(SEPT9_genotype) %in%
```

```
 colnames(SEPT9_rel_samples_ord)))
# 16 (PC9-477), 23 (PC9-29)


SEPT9_geno <- SEPT9_genotype[which(SEPT9_genotype$sample %in%
 colnames(SEPT9_rel_samples_ord)),]
identical(colnames(SEPT9_rel_samples_ord), rownames(SEPT9_geno))
 #TRUE
SEPT9_AA <- SEPT9_geno$sample[which(SEPT9_geno$kgp9468443 =="A/A")]
SEPT9_GA <- SEPT9_geno$sample[which(SEPT9_geno$kgp9468443 =="A/G")]
SEPT9_GG <- SEPT9_geno$sample[which(SEPT9_geno$kgp9468443 =="G/G")]
length(SEPT9_AA)  #16
length(SEPT9_GA)  #12
length(SEPT9_GG)  #9

SEPT9_AA_meth <- methLevel(SEPT9_rel_samples_ord)[, SEPT9_AA]
SEPT9_AA_medians <- rowMedians(SEPT9_AA_meth, na.rm=T)
SEPT9_GA_meth <- methLevel(SEPT9_rel_samples_ord)[, SEPT9_GA]
SEPT9_GA_medians <- rowMedians(SEPT9_GA_meth, na.rm=T)
SEPT9_GG_meth <- methLevel(SEPT9_rel_samples_ord)[, SEPT9_GG]
SEPT9_GG_medians <- rowMedians(SEPT9_GG_meth, na.rm=T)

png(file="/Users/ecazaly/Dropbox/Thesis/Chapter_5/Drafts/
 Genotype_methylation/from_LD/SEPT9_landscape.png", pointsize=12,
 units="mm", width=200, height=250, res=400)
par(mfrow=c(2,1), mar=c(5,3.3,4,1)+0.1, family="serif")
plot(start(ranges(SEPT9_rel_samples_ord))*10^-6, SEPT9_AA_medians,
 type="b", pch=15, col="cornflower blue", bg="green", xlab="",
 ylab="", main="Median Methylation by Genotype SEPT9
cg05161773 / rs426439", ylim=c(0,1.07))
title(ylab="Median Methylation", line=2.2, cex.lab=1.2)
title(xlab="Position (Mb)", line=2.2, cex.lab=1.2)
points(start(ranges(SEPT9_rel_samples_ord))*10^-6, type="b",
 SEPT9_GA_medians, pch=16, col="dark green")
points(start(ranges(SEPT9_rel_samples_ord))*10^-6, type="b",
 SEPT9_GG_medians, pch=17, col="hotpink")
abline(v=75378036*10^-6, col="black", lwd=1)
legend(75.378020, 1.13, legend="CpG-SNP: intronic in open sea",
 col="black", bty="n", cex=0.8)
legend(x=75.3776, y=0.3, pch=15, col="cornflower blue", bty="n",
 legend="AA (16)") #AA
legend(x= 75.3776, y=0.2, pch=16, col="dark green", bty="n",
 legend="GA (12)") #GA
legend(x= 75.3776, y=0.1, pch=17, col="hotpink", bty="n", legend="GG
 (9)") #GG
dev.off()
```

```
# MGMT:  cg09993319 / rs7898151 / chr10:131529435 / intronic in open
 sea
conversion[which(conversion$Name=="kgp8599981"),] #rs12571103
 131530307 - 131529435 = 872

# Use genotype at as this is ~1kb away. However this may not be in
 LD with CpG-SNP as haplotype block right on the site where the
 nearest SNP is.  The plot looks like it is though
which(colnames(gwaa_all_omni@gtdata)=="kgp8599981")
MGMT_gwaa <-
 gwaa_all_omni[,which(colnames(gwaa_all_omni@gtdata)=="kgp8599981")]
MGMT_genotype <- as.genotype.gwaa.data(MGMT_gwaa)
MGMT_genotype$sample <- rownames(MGMT_genotype)
MGMT_AA <- MGMT_genotype$sample[which(MGMT_genotype$kgp8599981 =="A/
 A")]
MGMT_GA <- MGMT_genotype$sample[which(MGMT_genotype$kgp8599981 =="A/
 G")]
MGMT_GG <- MGMT_genotype$sample[which(MGMT_genotype$kgp8599981 =="G/
 G")]
length(MGMT_AA)  #3
length(MGMT_GA)  #19
length(MGMT_GG)  #17
# there's 39 but should only be 37 as 2 biseq are bad quality.
 Perform below to eliminate and get geno data in right format

MGMT_detail <- regions[which(regions$Gene=="MGMT"),]
MGMT_raw <- clean_raw5_90nan[which(rownames(clean_raw5_90nan) %in%
 MGMT_detail$row.name),]
MGMT_rel <- rawToRel(MGMT_raw)

which(colnames(MGMT_rel) %in% rownames(MGMT_genotype))  #37
MGMT_rel_samples <- MGMT_rel[,which(colnames(MGMT_rel) %in%
 rownames(MGMT_genotype))]
colnames(MGMT_rel_samples)
rownames(MGMT_genotype) # not the same
MGMT_rel_samples_ord <-
 MGMT_rel_samples[,order(match(colnames(MGMT_rel_samples),
 rownames(MGMT_genotype)))]
colnames(MGMT_rel_samples_ord)
identical(colnames(MGMT_rel_samples_ord), rownames(MGMT_genotype))
 #FALSE, 2 geno samples not having seq data
which(!(rownames(MGMT_genotype) %in%
 colnames(MGMT_rel_samples_ord)))
# 16 (PC9-477), 23 (PC9-29)
```

```
MGMT_geno <- MGMT_genotype[which(MGMT_genotype$sample %in%
 colnames(MGMT_rel_samples_ord)),]
identical(colnames(MGMT_rel_samples_ord), rownames(MGMT_geno))
 #TRUE
MGMT_AA <- MGMT_geno$sample[which(MGMT_geno$kgp8599981 =="A/A")]
MGMT_GA <- MGMT_geno$sample[which(MGMT_geno$kgp8599981 =="A/G")]
MGMT_GG <- MGMT_geno$sample[which(MGMT_geno$kgp8599981 =="G/G")]
length(MGMT_AA)  #3
length(MGMT_GA)  #18
length(MGMT_GG)  #16

MGMT_AA_meth <- methLevel(MGMT_rel_samples_ord)[, MGMT_AA]
MGMT_AA_medians <- rowMedians(MGMT_AA_meth, na.rm=T)
MGMT_GA_meth <- methLevel(MGMT_rel_samples_ord)[, MGMT_GA]
MGMT_GA_medians <- rowMedians(MGMT_GA_meth, na.rm=T)
MGMT_GG_meth <- methLevel(MGMT_rel_samples_ord)[, MGMT_GG]
MGMT_GG_medians <- rowMedians(MGMT_GG_meth, na.rm=T)

png(file="/Users/ecazaly/Dropbox/Thesis/Chapter_5/Drafts/
 Genotype_methylation/from_LD/MGMT_landscape.png", pointsize=12,
 units="mm", width=200, height=250, res=400)
par(mfrow=c(2,1), mar=c(5,3.3,4,1)+0.1, family="serif")
plot(start(ranges(MGMT_rel_samples_ord))*10^-6, MGMT_AA_medians,
 type="b", pch=15, col="cornflower blue", bg="green", xlab="",
 ylab="", main="Median Methylation by Genotype MGMT
cg09993319 / rs7898151", ylim=c(0,1.07))
title(ylab="Median Methylation", line=2.2, cex.lab=1.2)
title(xlab="Position (Mb)", line=2.2, cex.lab=1.2)
points(start(ranges(MGMT_rel_samples_ord))*10^-6, type="b",
 MGMT_GA_medians, pch=16, col="dark green")
points(start(ranges(MGMT_rel_samples_ord))*10^-6, type="b",
 MGMT_GG_medians, pch=17, col="hotpink")
abline(v=131529435*10^-6, col="black", lwd=1)
legend(131529420*10^-6, 1.12, legend="CpG-SNP: intronic in open
 sea", col="black", bty="n", cex=0.8)
legend(x= 131.52948, y=0.3, pch=15, col="cornflower blue", bty="n",
 legend="AA (3)") #AA
legend(x= 131.52948, y=0.2, pch=16, col="dark green", bty="n",
 legend="GA (18)") #GA
legend(x= 131.52948, y=0.1, pch=17, col="hotpink", bty="n",
 legend="GG (16)") #GG
dev.off()
```

```
# MCC: cg08238375 / rs4705795 / chr5:112483149  / intronic in open
 sea
conversion[which(conversion$Name=="kgp1825521"),]  #rs57297544
 112482136  - 112483149 = -1013
# 112482033 -  112483149 = -1116
# 112483604 -  112483149 = 455   rs2416305

# Use genotype at rs2416305 as this is ~.5 kb away and the LD plot
 indicates in LD with CpG-SNP
which(colnames(gwaa_all_omni@gtdata)=="")

MCC_gwaa <-
 gwaa_all_omni[,which(colnames(gwaa_all_omni@gtdata)=="rs2416305")]
MCC_genotype <- as.genotype.gwaa.data(MCC_gwaa)
MCC_genotype$sample <- rownames(MCC_genotype)
MCC_AA <- MCC_genotype$sample[which(MCC_genotype$rs2416305 =="A/A")]
MCC_GA <- MCC_genotype$sample[which(MCC_genotype$rs2416305 =="A/G")]
MCC_GG <- MCC_genotype$sample[which(MCC_genotype$rs2416305 =="G/G")]
length(MCC_AA)  #29
length(MCC_GA)  #10
length(MCC_GG)  #0
# there's 39 but should only be 37 as 2 biseq are bad quality.
 Perform below to eliminate and get geno data in right format

MCC_detail <- regions[which(regions$Gene=="MCC"),]
MCC_raw <- clean_raw5_90nan[which(rownames(clean_raw5_90nan) %in%
 MCC_detail$row.name),]
MCC_rel <- rawToRel(MCC_raw)

which(colnames(MCC_rel) %in% rownames(MCC_genotype))  #37
MCC_rel_samples <- MCC_rel[,which(colnames(MCC_rel) %in%
 rownames(MCC_genotype))]
colnames(MCC_rel_samples)
rownames(MCC_genotype) # not the same
MCC_rel_samples_ord <-
 MCC_rel_samples[,order(match(colnames(MCC_rel_samples),
 rownames(MCC_genotype)))]
colnames(MCC_rel_samples_ord)
identical(colnames(MCC_rel_samples_ord), rownames(MCC_genotype))
 #FALSE, 2 geno samples not having seq data
which(!(rownames(MCC_genotype) %in% colnames(MCC_rel_samples_ord)))
# 16 (PC9-477), 23 (PC9-29)


MCC_geno <- MCC_genotype[which(MCC_genotype$sample %in%
 colnames(MCC_rel_samples_ord)),]
identical(colnames(MCC_rel_samples_ord), rownames(MCC_geno))  #TRUE
```

```
MCC_AA <- MCC_geno$sample[which(MCC_geno$rs2416305 =="A/A")]
MCC_GA <- MCC_geno$sample[which(MCC_geno$rs2416305 =="A/G")]
MCC_GG <- MCC_geno$sample[which(MCC_geno$rs2416305 =="G/G")]
length(MCC_AA)  # 27
length(MCC_GA)  # 10
length(MCC_GG)  # 0  only genotype to have 0 so far

MCC_AA_meth <- methLevel(MCC_rel_samples_ord)[, MCC_AA]
MCC_AA_medians <- rowMedians(MCC_AA_meth, na.rm=T)
MCC_GA_meth <- methLevel(MCC_rel_samples_ord)[, MCC_GA]
MCC_GA_medians <- rowMedians(MCC_GA_meth, na.rm=T)
MCC_GG_meth <- methLevel(MCC_rel_samples_ord)[, MCC_GG]
MCC_GG_medians <- rowMedians(MCC_GG_meth, na.rm=T)

png(file="/Users/ecazaly/Dropbox/Thesis/Chapter_5/Drafts/
 Genotype_methylation/from_LD/MCC_landscape.png", pointsize=12,
 units="mm", width=200, height=250, res=400)
par(mfrow=c(2,1), mar=c(5,3.3,4,1)+0.1, family="serif")
plot(start(ranges(MCC_rel_samples_ord))*10^-6, MCC_AA_medians,
 type="b", pch=15, col="cornflower blue", bg="green", xlab="",
 ylab="", main="Median Methylation by Genotype MCC
cg08238375 / rs4705795", ylim=c(0,1.07))
title(ylab="Median Methylation", line=2.2, cex.lab=1.2)
title(xlab="Position (Mb)", line=2.2, cex.lab=1.2)
points(start(ranges(MCC_rel_samples_ord))*10^-6, type="b",
 MCC_GA_medians, pch=16, col="dark green")
#points(start(ranges(MCC_rel_samples_ord))*10^-6, type="b",
 MCC_GG_medians, pch=17, col="hotpink")
abline(v=112483149*10^-6, col="black", lwd=1)
legend(112.48314, 1.12, legend="CpG-SNP: intronic in open sea",
 col="black", bty="n", cex=0.8)
legend(x=112.48323, y=0.3, pch=15, col="cornflower blue", bty="n",
 legend="AA (27)") #AA
legend(x=112.48323, y=0.2, pch=16, col="dark green", bty="n",
 legend="GA (10)") #GA
legend(x=112.48323, y=0.1, pch=17, col="hotpink", bty="n",
 legend="GG (0)") #GG
dev.off()
```

```
#################################################################
## 3. Examine methylation levels between cancer/control groups ##
```

```
#################################################################


# Methylation Landscape Plots --> colour methylation values by case/
control status

# Clean data generated in above section
clean_raw5_90nan
clean_rel5_90nan

cancer <- clean_raw5_90nan[, which(clean_raw5_90nan@colData
 $Disease=="A")]
# 248, 31
control <- clean_raw5_90nan[, which(clean_raw5_90nan@colData
 $Disease=="DVA_control")]
# 248, 32

# subset this data by meQTL region
AJAP_raw <- clean_raw5_90nan[which(rownames(clean_raw5_90nan) %in%
 AJAP_detail$row.name),]
AJAP_rel <- rawToRel(AJAP_raw)
CASZ1_raw <- clean_raw5_90nan[which(rownames(clean_raw5_90nan) %in%
 CASZ1_detail$row.name),]
CASZ1_rel <- rawToRel(CASZ1_raw)
NME6_raw <- clean_raw5_90nan[which(rownames(clean_raw5_90nan) %in%
 NME6_detail$row.name),]
NME6_rel <- rawToRel(NME6_raw)
MCC_raw <- clean_raw5_90nan[which(rownames(clean_raw5_90nan) %in%
 MCC_detail$row.name),]
MCC_rel <- rawToRel(MCC_raw)
C10orf46_raw <- clean_raw5_90nan[which(rownames(clean_raw5_90nan)
 %in% C10orf46_detail$row.name),]
C10orf46_rel <- rawToRel(C10orf46_raw)
MGMT_raw <- clean_raw5_90nan[which(rownames(clean_raw5_90nan) %in%
 MGMT_detail$row.name),]
MGMT_rel <- rawToRel(MGMT_raw)
USP7_raw <- clean_raw5_90nan[which(rownames(clean_raw5_90nan) %in%
 USP7_detail$row.name),]
USP7_rel <- rawToRel(USP7_raw)
PRM1_raw <- clean_raw5_90nan[which(rownames(clean_raw5_90nan) %in%
 PRM1_detail$row.name),]
PRM1_rel <- rawToRel(PRM1_raw)
FOXK2_raw <- clean_raw5_90nan[which(rownames(clean_raw5_90nan) %in%
 FOXK2_detail$row.name),]
FOXK2_rel <- rawToRel(FOXK2_raw)
SEPT9_raw <- clean_raw5_90nan[which(rownames(clean_raw5_90nan) %in%
 SEPT9_detail$row.name),]
```

```
SEPT9_rel <- rawToRel(SEPT9_raw)
RAB11_raw <- clean_raw5_90nan[which(rownames(clean_raw5_90nan) %in%
 RAB11_detail$row.name),]
RAB11_rel <- rawToRel(RAB11_raw)
ITGB2_raw <- clean_raw5_90nan[which(rownames(clean_raw5_90nan) %in%
 ITGB2_detail$row.name),]
ITGB2_rel <- rawToRel(ITGB2_raw)

# AJAP
median_AJAP_cancer <- rowMedians(methLevel(AJAP_rel[,
 which(AJAP_rel@colData$Disease=="A")]), na.rm=T)
median_AJAP_control <- rowMedians(methLevel(AJAP_rel[,
 which(AJAP_rel@colData$Disease=="DVA_control")]), na.rm=T)
png(file="/Users/ecazaly/Dropbox/Thesis/Chapter_5/Drafts/
 AJAP_cancer.png", pointsize=12, units="mm", width=200, height=250,
 res=400)
par(mfrow=c(2,1), mar=c(5,3.3,4,1)+0.1)  #c(bottom, left, top,
 right), default is c(5, 4, 4, 2) + 0.1.
plot(rep(start(ranges(AJAP_rel))*10^-6,ncol(cancer)),
 methLevel(AJAP_rel[, which(AJAP_rel@colData$Disease=="A")]),
 type="p", pch=16, cex=0.5, col="red", xlab="", ylab="", main="AJAP:
 Individual Methylation Levels", ylim=c(0,1.1))
title(ylab="Methylation Level", line=2.2, cex.lab=1.2)
title(xlab="Position (Mb)", line=2.2, cex.lab=1.2)
points(rep(start(ranges(AJAP_rel))*10^-6, ncol(control)),
 methLevel(AJAP_rel[, which(AJAP_rel@colData
 $Disease=="DVA_control")]), pch=17, col="blue", cex=0.5)
points(rep(start(ranges(AJAP_rel))*10^-6), median_AJAP_cancer,
 type="o", pch=16, cex=0.5, col="red", xlab="Genomic Position (Mb)",
 ylab="Methylation", main="Median Methylation at AJAP by cancer
 status")
points(rep(start(ranges(AJAP_rel))*10^-6), median_AJAP_control,
 pch=17, col="blue", cex=0.5, type="o")
abline(v=4725584*10^-6, col="black", lwd=1.5)
legend(x=(4725584*10^-6)-0.00004, y=1.16, legend="CpG-SNP: intronic
 in CpG Shore", bty="n", cex=.8)
legend(x=(start(ranges(AJAP_rel))[1]*10^-6)+0.00013,y=1.17, pch=16,
 col="red", legend="Cancer", cex=1, bty="n")
legend(x=(start(ranges(AJAP_rel))[1]*10^-6)-0.00006, y=1.17, pch=17,
 col="blue", legend="Control", cex=1, bty="n")
plot(rep(start(ranges(AJAP_rel))*10^-6), median_AJAP_cancer,
 type="o", pch=16, cex=0.5, col="red", xlab="", ylab="", main="AJAP:
 Median Methylation per Group", ylim=c(0,1.1))
title(ylab="Methylation Level", line=2.2, cex.lab=1.2)
title(xlab="Position (Mb)", line=2.2, cex.lab=1.2)
points(rep(start(ranges(AJAP_rel))*10^-6), median_AJAP_control,
 pch=17, col="blue", cex=0.5, type="o")
```

```
abline(v=4725584*10^-6, col="black", lwd=1.5)
legend(x=(4725584*10^-6)-0.00004, y=1.16, legend="CpG-SNP; intronic
 in CpG Shore", bty="n", cex=.8)
legend(x=(start(ranges(AJAP_rel))[1]*10^-6)+0.00013,y=1.17, pch=16,
 col="red", legend="Cancer", cex=1, bty="n")
legend(x=(start(ranges(AJAP_rel))[1]*10^-6)-0.00006, y=1.17, pch=17,
 col="blue", legend="Control", cex=1, bty="n")
dev.off()


# CASZ1
png(file="/Users/ecazaly/Dropbox/Thesis/Chapter_5/Drafts/
 CASZ1_cancer.png", pointsize=12, units="mm", width=200, height=250,
 res=400)
par(mfrow=c(2,1), mar=c(5,3.3,4,1)+0.1)
plot(rep(start(ranges(CASZ1_rel))*10^-6,ncol(cancer)),
 methLevel(CASZ1_rel[, which(CASZ1_raw@colData$Disease=="A")]),
 type="p", pch=16, cex=0.5, col="red", xlab="", ylab="",
 main="CASZ1: Individual Methylation Levels", ylim=c(0,1.1))
title(ylab="Methylation Level", line=2.2, cex.lab=1.2)
title(xlab="Position (Mb)", line=2.2, cex.lab=1.2)
points(rep(start(ranges(CASZ1_rel))*10^-6, ncol(control)),
 methLevel(CASZ1_rel[, which(CASZ1_raw@colData
 $Disease=="DVA_control")]), pch=17, col="blue", cex=0.5)
points(rep(start(ranges(CASZ1_rel))*10^-6), median_CASZ1_cancer,
 type="o", pch=16, cex=0.5, col="red", xlab="Genomic Position (Mb)",
 ylab="Methylation", main="Median Methylation at CASZ1 by cancer
 status")
points(rep(start(ranges(CASZ1_rel))*10^-6), median_CASZ1_control,
 pch=17, col="blue", cex=0.5, type="o")
abline(v=10737562*10^-6, col="black", lwd=1)
legend(x=(10737562*10^-6)-0.00027, y=1.16, legend="CpG-SNP; intronic
 in open sea", bty="n", cex=.8)
legend(x=(start(ranges(CASZ1_rel))[1]*10^-6)+0.0001,y=1.17, pch=16,
 col="red", legend="Cancer", cex=1, bty="n")
legend(x=(start(ranges(CASZ1_rel))[1]*10^-6)-0.000009, y=1.17,
 pch=17, col="blue", legend="Control", cex=1, bty="n")
median_CASZ1_cancer <- rowMedians(methLevel(CASZ1_rel[,
 which(CASZ1_raw@colData$Disease=="A")]), na.rm=T)
median_CASZ1_control <- rowMedians(methLevel(CASZ1_rel[,
 which(CASZ1_raw@colData$Disease=="DVA_control")]), na.rm=T)
plot(rep(start(ranges(CASZ1_rel))*10^-6), median_CASZ1_cancer,
 type="o", pch=16, cex=0.5, col="red", xlab="", ylab="",
 main="CASZ1: Median Methylation per Group", ylim=c(0,1.1))
title(ylab="Methylation Level", line=2.2, cex.lab=1.2)
title(xlab="Position (Mb)", line=2.2, cex.lab=1.2)
points(rep(start(ranges(CASZ1_rel))*10^-6), median_CASZ1_control,
```

```
 pch=17, col="blue", cex=0.5, type="o")
abline(v=10737562*10^-6, col="black", lwd=1)
legend(x=(10737562*10^-6)-0.00027, y=1.16, legend="CpG-SNP; intronic
 in open sea", bty="n", cex=.8)
legend(x=(start(ranges(CASZ1_rel))[1]*10^-6)+0.0001,y=1.17, pch=16,
 col="red", legend="Cancer", cex=1, bty="n")
legend(x=(start(ranges(CASZ1_rel))[1]*10^-6)-0.000009, y=1.17,
 pch=17, col="blue", legend="Control", cex=1, bty="n")
dev.off()


# PRM1
median_PRM1_cancer <- rowMedians(methLevel(PRM1_rel[,
 which(PRM1_raw@colData$Disease=="A")]), na.rm=T)
median_PRM1_control <- rowMedians(methLevel(PRM1_rel[,
 which(PRM1_raw@colData$Disease=="DVA_control")]), na.rm=T)
png(file="/Users/ecazaly/Dropbox/Thesis/Chapter_5/Drafts/
 PRM1_cancer.png", pointsize=12, units="mm", width=200, height=250,
 res=400)
par(mfrow=c(2,1), mar=c(5,3.3,4,1)+0.1)
plot(rep(start(ranges(PRM1_rel))*10^-6,ncol(cancer)),
 methLevel(PRM1_rel[, which(PRM1_raw@colData$Disease=="A")]),
 type="p", pch=16, cex=0.5, col="red", xlab="", ylab="", main="PRM1:
 Individual Methylation Levels", ylim=c(0,1.1))
title(ylab="Methylation Level", line=2.2, cex.lab=1.2)
title(xlab="Position (Mb)", line=2.2, cex.lab=1.2)
points(rep(start(ranges(PRM1_rel))*10^-6, ncol(control)),
 methLevel(PRM1_rel[, which(PRM1_raw@colData
 $Disease=="DVA_control")]), pch=17, col="blue", cex=0.5)
points(rep(start(ranges(PRM1_rel))*10^-6), median_PRM1_cancer,
 type="o", pch=16, cex=0.5, col="red", xlab="Genomic Position (Mb)",
 ylab="Methylation", main="Median Methylation at PRM1 by cancer
 status", ylim=c(0,1.1))
points(rep(start(ranges(PRM1_rel))*10^-6), median_PRM1_control,
 pch=17, col="blue", cex=0.5, type="o")
abline(v=11374865*10^-6, col="black", lwd=1)
legend(x=(11374865*10^-6)+0.000006, y=1.16, legend="CpG-SNP: coding
 (synonymous) in open sea", bty="n", cex=.8)
legend(x=(start(ranges(PRM1_rel))[1]*10^-6)+0.00014,y=1.17, pch=16,
 col="red", legend="Cancer", cex=1, bty="n")
legend(x=(start(ranges(PRM1_rel))[1]*10^-6)-0.00002, y=1.17, pch=17,
 col="blue", legend="Control", cex=1, bty="n")
plot(rep(start(ranges(PRM1_rel))*10^-6), median_PRM1_cancer,
 type="o", pch=16, cex=0.5, col="red", xlab="", ylab="", main="PRM1:
 Median Methylation by Group", ylim=c(0,1.1))
title(ylab="Methylation Level", line=2.2, cex.lab=1.2)
title(xlab="Position (Mb)", line=2.2, cex.lab=1.2)
```

```
points(rep(start(ranges(PRM1_rel))*10^-6), median_PRM1_control,
 pch=17, col="blue", cex=0.5, type="o")
abline(v=11374865*10^-6, col="black", lwd=1)
legend(x=(11374865*10^-6)+0.000006, y=1.16, legend="CpG-SNP: coding
 (synonymous) in open sea", bty="n", cex=.8)
legend(x=(start(ranges(PRM1_rel))[1]*10^-6)+0.00014,y=1.17, pch=16,
 col="red", legend="Cancer", cex=1, bty="n")
legend(x=(start(ranges(PRM1_rel))[1]*10^-6)-0.00002, y=1.17, pch=17,
 col="blue", legend="Control", cex=1, bty="n")
dev.off()


# USP7
median_USP7_cancer <- rowMedians(methLevel(USP7_rel[,
 which(USP7_raw@colData$Disease=="A")]), na.rm=T)
median_USP7_control <- rowMedians(methLevel(USP7_rel[,
 which(USP7_raw@colData$Disease=="DVA_control")]), na.rm=T)
png(file="/Users/ecazaly/Dropbox/Thesis/Chapter_5/Drafts/
 USP7_cancer.png", pointsize=12, units="mm", width=200, height=250,
 res=400)
par(mfrow=c(2,1), mar=c(5,3.3,4,1)+0.1)
plot(rep(start(ranges(USP7_rel))*10^-6,ncol(cancer)),
 methLevel(USP7_rel[, which(USP7_raw@colData$Disease=="A")]),
 type="p", pch=16, cex=0.5, col="red", xlab="", ylab="", main="USP7:
 Individual Methylation Levels", ylim=c(0,1.1))
title(ylab="Methylation Level", line=2.2, cex.lab=1.2)
title(xlab="Position (Mb)", line=2.2, cex.lab=1.2)
points(rep(start(ranges(USP7_rel))*10^-6, ncol(control)),
 methLevel(USP7_rel[, which(USP7_raw@colData
 $Disease=="DVA_control")]), pch=17, col="blue", cex=0.5)
points(rep(start(ranges(USP7_rel))*10^-6), median_USP7_cancer,
 type="o", pch=16, cex=0.5, col="red", xlab="Genomic Position (Mb)",
 ylab="Methylation", main="Median Methylation at USP7 by cancer
 status")
points(rep(start(ranges(USP7_rel))*10^-6), median_USP7_control,
 pch=17, col="blue", cex=0.5, type="o")
abline(v=8995926*10^-6, col="black", lwd=1)
legend(x=(8995926*10^-6)-0.00025, y=1.16, legend="CpG-SNP:
 intragenic in open sea", bty="n", cex=.8)
legend(x=(start(ranges(USP7_rel))[1]*10^-6)+0.0001,y=1.17, pch=16,
 col="red", legend="Cancer", cex=1, bty="n")
legend(x=(start(ranges(USP7_rel))[1]*10^-6)-0.00001, y=1.17, pch=17,
 col="blue", legend="Control", cex=1, bty="n")
plot(rep(start(ranges(USP7_rel))*10^-6), median_USP7_cancer,
 type="o", pch=16, cex=0.5, col="red", xlab="", ylab="", main="USP7:
 Median Methylation per Group", ylim=c(0,1.1))
title(ylab="Methylation Level", line=2.2, cex.lab=1.2)
```

```
title(xlab="Position (Mb)", line=2.2, cex.lab=1.2)
points(rep(start(ranges(USP7_rel))*10^-6), median_USP7_control,
 pch=17, col="blue", cex=0.5, type="o")
abline(v=8995926*10^-6, col="black", lwd=1)
legend(x=(8995926*10^-6)-0.00025, y=1.16, legend="CpG-SNP:
 intragenic in open sea", bty="n", cex=.8)
legend(x=(start(ranges(USP7_rel))[1]*10^-6)+0.0001,y=1.17, pch=16,
 col="red", legend="Cancer", cex=1, bty="n")
legend(x=(start(ranges(USP7_rel))[1]*10^-6)-0.00001, y=1.17, pch=17,
 col="blue", legend="Control", cex=1, bty="n")
dev.off()


# FOXK2
median_FOXK2_cancer <- rowMedians(methLevel(FOXK2_rel[,
 which(FOXK2_raw@colData$Disease=="A")]), na.rm=T)
median_FOXK2_control <- rowMedians(methLevel(FOXK2_rel[,
 which(FOXK2_raw@colData$Disease=="DVA_control")]), na.rm=T)
png(file="/Users/ecazaly/Dropbox/Thesis/Chapter_5/Drafts/
 FOXK2_cancer.png", pointsize=12, units="mm", width=200, height=250,
 res=400)
par(mfrow=c(2,1), mar=c(5,3.3,4,1)+0.1)
plot(rep(start(ranges(FOXK2_rel))*10^-6,ncol(cancer)),
 methLevel(FOXK2_rel[, which(FOXK2_raw@colData$Disease=="A")]),
 type="p", pch=16, cex=0.5, col="red", xlab="", ylab="",
 main="FOXK2: Individual Methylation Levels", ylim=c(0,1.1))
title(ylab="Methylation Level", line=2.2, cex.lab=1.2)
title(xlab="Position (Mb)", line=2.2, cex.lab=1.2)
points(rep(start(ranges(FOXK2_rel))*10^-6, ncol(control)),
 methLevel(FOXK2_rel[, which(FOXK2_raw@colData
 $Disease=="DVA_control")]), pch=17, col="blue", cex=0.5)
abline(v=80535367*10^-6, col="black", lwd=1)
legend(x=(80535367*10^-6)+0.0003, y=1.16, legend="CpG-SNP: intronic
 in CpG shore", bty="n", cex=.8)
points(rep(start(ranges(FOXK2_rel))*10^-6), median_FOXK2_cancer,
 type="o", pch=16, cex=0.5, col="red", xlab="Genomic Position (Mb)",
 ylab="Methylation", main="Median Methylation at FOXK2 by cancer
 status", ylim=c(0,1.1))
points(rep(start(ranges(FOXK2_rel))*10^-6), median_FOXK2_control,
 pch=17, col="blue", cex=0.5, type="o")
legend(x=(start(ranges(FOXK2_rel))[1]*10^-6)+0.00013,y=1.17, pch=16,
 col="red", legend="Cancer", cex=1, bty="n")
legend(x=(start(ranges(FOXK2_rel))[1]*10^-6)+0.00003, y=1.17,
 pch=17, col="blue", legend="Control", cex=1, bty="n")
plot(rep(start(ranges(FOXK2_rel))*10^-6), median_FOXK2_cancer,
 type="o", pch=16, cex=0.5, col="red", xlab="", ylab="",
 main="ITBG2: Median Methylation by Group", ylim=c(0,1.1))
```

```
title(ylab="Methylation Level", line=2.2, cex.lab=1.2)
title(xlab="Position (Mb)", line=2.2, cex.lab=1.2)
points(rep(start(ranges(FOXK2_rel))*10^-6), median_FOXK2_control,
 pch=17, col="blue", cex=0.5, type="o")
abline(v=80535367*10^-6, col="black", lwd=1)
legend(x=(80535367*10^-6)+0.0003, y=1.16, legend="CpG-SNP: intronic
 in CpG shore", bty="n", cex=.8)
legend(x=(start(ranges(FOXK2_rel))[1]*10^-6)+0.00013,y=1.17, pch=16,
 col="red", legend="Cancer", cex=1, bty="n")
legend(x=(start(ranges(FOXK2_rel))[1]*10^-6)+0.00003, y=1.17,
 pch=17, col="blue", legend="Control", cex=1, bty="n")
dev.off()


# NME6
median_NME6_cancer <- rowMedians(methLevel(NME6_rel[,
 which(NME6_raw@colData$Disease=="A")]), na.rm=T)
median_NME6_control <- rowMedians(methLevel(NME6_rel[,
 which(NME6_raw@colData$Disease=="DVA_control")]), na.rm=T)
png(file="/Users/ecazaly/Dropbox/Thesis/Chapter_5/Drafts/
 NME6_cancer.png", pointsize=12, units="mm", width=200, height=250,
 res=400)
par(mfrow=c(2,1), mar=c(5,3.3,4,1)+0.1)
plot(rep(start(ranges(NME6_rel))*10^-6,ncol(cancer)),
 methLevel(NME6_rel[, which(NME6_raw@colData$Disease=="A")]),
 type="p", pch=16, cex=0.5, col="red", xlab="", ylab="", main="NME6:
 Individual Methylation Level", ylim=c(0,1.1))
title(ylab="Methylation Level", line=2.2, cex.lab=1.2)
title(xlab="Position (Mb)", line=2.2, cex.lab=1.2)
points(rep(start(ranges(NME6_rel))*10^-6, ncol(control)),
 methLevel(NME6_rel[, which(NME6_raw@colData
 $Disease=="DVA_control")]), pch=17, col="blue", cex=0.5)
points(rep(start(ranges(NME6_rel))*10^-6), median_NME6_cancer,
 type="o", pch=16, cex=0.5, col="red", xlab="Position (Mb)",
 ylab="methylation", main="Median Methylation at NME6 by cancer
 status")
points(rep(start(ranges(NME6_rel))*10^-6), median_NME6_control,
 pch=17, col="blue", cex=0.5, type="o")
legend(x=(start(ranges(NME6_rel))[1]*10^-6)+0.00008,y=1.17, pch=16,
 col="red", legend="Cancer", cex=1, bty="n")
legend(x=(start(ranges(NME6_rel))[1]*10^-6)-0.00001, y=1.17, pch=17,
 col="blue", legend="Control", cex=1, bty="n")
abline(v=48335857*10^-6, col="black", lwd=1)
legend(x=(48335857*10^-6)-0.000009, y=1.16, legend="CpG-SNP: UTR in
 open sea", bty="n", cex=.8)
plot(rep(start(ranges(NME6_rel))*10^-6), median_NME6_cancer,
 type="o", pch=16, cex=0.5, col="red", xlab="", ylab="", main="NME6:
```

```
 Median Methylation by Group", ylim=c(0,1.1))
title(ylab="Methylation Level", line=2.2, cex.lab=1.2)
title(xlab="Position (Mb)", line=2.2, cex.lab=1.2)
points(rep(start(ranges(NME6_rel))*10^-6), median_NME6_control,
 pch=17, col="blue", cex=0.5, type="o")
legend(x=(start(ranges(NME6_rel))[1]*10^-6)+0.00008,y=1.17, pch=16,
 col="red", legend="Cancer", cex=1, bty="n")
legend(x=(start(ranges(NME6_rel))[1]*10^-6)-0.00001, y=1.17, pch=17,
 col="blue", legend="Control", cex=1, bty="n")
abline(v=48335857*10^-6, col="black", lwd=1)
legend(x=(48335857*10^-6)-0.000009, y=1.16, legend="CpG-SNP: UTR in
 open sea", bty="n", cex=.8)
dev.off()


# MCC
median_MCC_cancer <- rowMedians(methLevel(MCC_rel[,
 which(MCC_raw@colData$Disease=="A")]), na.rm=T)
median_MCC_control <- rowMedians(methLevel(MCC_rel[,
 which(MCC_raw@colData$Disease=="DVA_control")]), na.rm=T)
png(file="/Users/ecazaly/Dropbox/Thesis/Chapter_5/Drafts/
 MCC_cancer.png", pointsize=12, units="mm", width=200, height=250,
 res=400)
par(mfrow=c(2,1), mar=c(5,3.3,4,1)+0.1)
plot(rep(start(ranges(MCC_rel))*10^-6,ncol(cancer)),
 methLevel(MCC_rel[, which(MCC_raw@colData$Disease=="A")]),
 type="p", pch=16, cex=0.5, col="red", xlab="", ylab="", main="MCC:
 Individual Methylation Level", ylim=c(0,1.1), xlim=c(112.4830999,
 112.4837))
title(ylab="Methylation Level", line=2.2, cex.lab=1.2)
title(xlab="Position (Mb)", line=2.2, cex.lab=1.2)
points(rep(start(ranges(MCC_rel))*10^-6, ncol(control)),
 methLevel(MCC_rel[, which(MCC_raw@colData
 $Disease=="DVA_control")]), pch=17, col="blue", cex=0.5)
points(rep(start(ranges(MCC_rel))*10^-6), median_MCC_cancer,
 type="o", pch=16, cex=0.5, col="red", xlab="Position (Mb)",
 ylab="methylation", main="Median Methylation at MCC by cancer
 status")
points(rep(start(ranges(MCC_rel))*10^-6), median_MCC_control,
 pch=17, col="blue", cex=0.5, type="o")
abline(v=112483149*10^-6, col="black", lwd=1)
legend(x=(112483149*10^-6)+0.0003, y=1.16, legend="CpG-SNP: intronic
 in open sea", bty="n", cex=.8)
legend(x=(start(ranges(MCC_rel))[1]*10^-6)+0.00013,y=1.17, pch=16,
 col="red", legend="Cancer", cex=1, bty="n")
legend(x=(start(ranges(MCC_rel))[1]*10^-6)+0.00005, y=1.17, pch=17,
 col="blue", legend="Control", cex=1, bty="n")
```

```
plot(rep(start(ranges(MCC_rel))*10^-6), median_MCC_cancer, type="o",
 pch=16, cex=0.5, col="red", xlab="", ylab="", main="MCC: Median
 Methylation by Group", ylim=c(0,1.1), xlim=c(112.4830999,
 112.4837))
title(ylab="Methylation Level", line=2.2, cex.lab=1.2)
title(xlab="Position (Mb)", line=2.2, cex.lab=1.2)
points(rep(start(ranges(MCC_rel))*10^-6), median_MCC_control,
 pch=17, col="blue", cex=0.5, type="o")
abline(v=112483149*10^-6, col="black", lwd=1)
legend(x=(112483149*10^-6)+0.0003, y=1.16, legend="CpG-SNP: intronic
 in open sea", bty="n", cex=.8)
legend(x=(start(ranges(MCC_rel))[1]*10^-6)+0.00013,y=1.17, pch=16,
 col="red", legend="Cancer", cex=1, bty="n")
legend(x=(start(ranges(MCC_rel))[1]*10^-6)+0.00005, y=1.17, pch=17,
 col="blue", legend="Control", cex=1, bty="n")
dev.off()


# C10orf46
median_C10orf46_cancer <- rowMedians(methLevel(C10orf46_rel[,
 which(C10orf46_rel@colData$Disease=="A")]), na.rm=T)
median_C10orf46_control <- rowMedians(methLevel(C10orf46_rel[,
 which(C10orf46_rel@colData$Disease=="DVA_control")]), na.rm=T)
png(file="/Users/ecazaly/Dropbox/Thesis/Chapter_5/Drafts/
 C10orf46_cancer.png", pointsize=12, units="mm", width=200,
 height=250, res=400)
par(mfrow=c(2,1), mar=c(5,3.3,4,1)+0.1)
plot(rep(start(ranges(C10orf46_rel))*10^-6,ncol(cancer)),
 methLevel(C10orf46_rel[, which(C10orf46_rel@colData
 $Disease=="A")]), type="p", pch=16, cex=0.5, col="red", xlab="",
 ylab="", main="C10orf46: Individual Methylation Levels",
 ylim=c(0,1.1))
title(ylab="Methylation Level", line=2.2, cex.lab=1.2)
title(xlab="Position (Mb)", line=2.2, cex.lab=1.2)
points(rep(start(ranges(C10orf46_rel))*10^-6, ncol(control)),
 methLevel(C10orf46_rel[, which(C10orf46_raw@colData
 $Disease=="DVA_control")]), pch=17, col="blue", cex=0.5)
abline(v=120516119*10^-6, col="black", lwd=1)
legend(x=(120516119*10^-6)-0.00025, y=1.16, legend="CpG-SNP;
 intronic in CpG shore", bty="n", cex=.8)
points(rep(start(ranges(C10orf46_rel))*10^-6),
 median_C10orf46_cancer, type="o", pch=16, cex=0.5, col="red",
 xlab="Genomic Position (Mb)", ylab="Methylation", main="Median
 Methylation at C10orf46 by cancer status")
points(rep(start(ranges(C10orf46_rel))*10^-6),
 median_C10orf46_control, pch=17, col="blue", cex=0.5, type="o")
legend(x=(start(ranges(C10orf46_rel))[1]*10^-6)+0.0001,y=1.17,
```

```
 pch=16, col="red", legend="Cancer", cex=1, bty="n")
legend(x=(start(ranges(C10orf46_rel))[1]*10^-6)-0.00002, y=1.17,
 pch=17, col="blue", legend="Control", cex=1, bty="n")
plot(rep(start(ranges(C10orf46_rel))*10^-6), median_C10orf46_cancer,
 type="o", pch=16, cex=0.5, col="red", xlab="", ylab="",
 main="C10orf46: Median Methylation per Group", ylim=c(0,1.1))
title(ylab="Methylation Level", line=2.2, cex.lab=1.2)
title(xlab="Position (Mb)", line=2.2, cex.lab=1.2)
points(rep(start(ranges(C10orf46_rel))*10^-6),
 median_C10orf46_control, pch=17, col="blue", cex=0.5, type="o")
abline(v=120516119*10^-6, col="black", lwd=1)
legend(x=(120516119*10^-6)-0.00025, y=1.16, legend="CpG-SNP;
 intronic in CpG shore", bty="n", cex=.8)
legend(x=(start(ranges(C10orf46_rel))[1]*10^-6)+0.0001,y=1.17,
 pch=16, col="red", legend="Cancer", cex=1, bty="n")
legend(x=(start(ranges(C10orf46_rel))[1]*10^-6)-0.00002, y=1.17,
 pch=17, col="blue", legend="Control", cex=1, bty="n")
dev.off()


## MGMT
median_MGMT_cancer <- rowMedians(methLevel(MGMT_rel[,
 which(MGMT_raw@colData$Disease=="A")]), na.rm=T)
median_MGMT_control <- rowMedians(methLevel(MGMT_rel[,
 which(MGMT_raw@colData$Disease=="DVA_control")]), na.rm=T)
png(file="/Users/ecazaly/Dropbox/Thesis/Chapter_5/Drafts/
 MGMT_cancer.png", pointsize=12, units="mm", width=200, height=250,
 res=400)
par(mfrow=c(2,1), mar=c(5,3.3,4,1)+0.1)
plot(rep(start(ranges(MGMT_rel))*10^-6,ncol(cancer)),
 methLevel(MGMT_rel[, which(MGMT_raw@colData$Disease=="A")]),
 type="p", pch=16, cex=0.5, col="red", xlab="", ylab="", main="MGMT:
 Individual Methylation Level", ylim=c(0,1.1))
title(ylab="Methylation Level", line=2.2, cex.lab=1.2)
title(xlab="Position (Mb)", line=2.2, cex.lab=1.2)
points(rep(start(ranges(MGMT_rel))*10^-6, ncol(control)),
 methLevel(MGMT_rel[, which(MGMT_raw@colData
 $Disease=="DVA_control")]), pch=17, col="blue", cex=0.5)
abline(v=131529435*10^-6, col="black", lwd=1)
legend(x=(131529435*10^-6)+0.0003, y=1.16, legend="CpG-SNP: intronic
 in open sea", bty="n", cex=.8)
points(rep(start(ranges(MGMT_rel))*10^-6), median_MGMT_cancer,
 type="o", pch=16, cex=0.5, col="red", xlab="Genomic Position (Mb)",
 ylab="Methylation", main="Median Methylation at MGMT by cancer
 status")
points(rep(start(ranges(MGMT_rel))*10^-6), median_MGMT_control,
 pch=17, col="blue", cex=0.5, type="o")
```

```
legend(x=(start(ranges(MGMT_rel))[1]*10^-6)+0.0002,y=1.17, pch=16,
 col="red", legend="Cancer", cex=1, bty="n")
legend(x=(start(ranges(MGMT_rel))[1]*10^-6)+0.000115, y=1.17,
 pch=17, col="blue", legend="Control", cex=1, bty="n")
plot(rep(start(ranges(MGMT_rel))*10^-6), median_MGMT_cancer,
 type="o", pch=16, cex=0.5, col="red", xlab="", ylab="", main="MGMT:
 Median Methylation by Group", ylim=c(0,1.1))
title(ylab="Methylation Level", line=2.2, cex.lab=1.2)
title(xlab="Position (Mb)", line=2.2, cex.lab=1.2)
points(rep(start(ranges(MGMT_rel))*10^-6), median_MGMT_control,
 pch=17, col="blue", cex=0.5, type="o")
abline(v=131529435*10^-6, col="black", lwd=1)
legend(x=(131529435*10^-6)+0.0003, y=1.16, legend="CpG-SNP: intronic
 in open sea", bty="n", cex=.8)
legend(x=(start(ranges(MGMT_rel))[1]*10^-6)+0.0002,y=1.17, pch=16,
 col="red", legend="Cancer", cex=1, bty="n")
legend(x=(start(ranges(MGMT_rel))[1]*10^-6)+0.000115, y=1.17,
 pch=17, col="blue", legend="Control", cex=1, bty="n")
dev.off()


# SEPT9
median_SEPT9_cancer <- rowMedians(methLevel(SEPT9_rel[,
 which(SEPT9_raw@colData$Disease=="A")]), na.rm=T)
median_SEPT9_control <- rowMedians(methLevel(SEPT9_rel[,
 which(SEPT9_raw@colData$Disease=="DVA_control")]), na.rm=T)
png(file="/Users/ecazaly/Dropbox/Thesis/Chapter_5/Drafts/
 SEPT9_cancer.png", pointsize=12, units="mm", width=200, height=250,
 res=400)
par(mfrow=c(2,1), mar=c(5,3.3,4,1)+0.1)
plot(rep(start(ranges(SEPT9_rel))*10^-6,ncol(cancer)),
 methLevel(SEPT9_rel[, which(SEPT9_raw@colData$Disease=="A")]),
 type="p", pch=16, cex=0.5, col="red", xlab="", ylab="",
 main="SEPT9: Individual Methylation Level", ylim=c(0,1.1))
title(ylab="Methylation Level", line=2.2, cex.lab=1.2)
title(xlab="Position (Mb)", line=2.2, cex.lab=1.2)
points(rep(start(ranges(SEPT9_rel))*10^-6, ncol(control)),
 methLevel(SEPT9_rel[, which(SEPT9_raw@colData
 $Disease=="DVA_control")]), pch=17, col="blue", cex=0.5)
abline(v=75378036*10^-6, col="black", lwd=1)
legend(x=(75378036*10^-6)-0.000015, y=1.16, legend="CpG-SNP:
 intronic in open sea", bty="n", cex=.8)
points(rep(start(ranges(SEPT9_rel))*10^-6), median_SEPT9_cancer,
 type="o", pch=16, cex=0.5, col="red", xlab="Genomic Position (Mb)",
 ylab="Methylation", main="Median Methylation at SEPT9 by cancer
 status")
points(rep(start(ranges(SEPT9_rel))*10^-6), median_SEPT9_control,
```

```
 pch=17, col="blue", cex=0.5, type="o")
legend(x=(start(ranges(SEPT9_rel))[1]*10^-6)+0.00012,y=1.17, pch=16,
 col="red", legend="Cancer", cex=1, bty="n")
legend(x=(start(ranges(SEPT9_rel))[1]*10^-6)-0.00002, y=1.17,
 pch=17, col="blue", legend="Control", cex=1, bty="n")
plot(rep(start(ranges(SEPT9_rel))*10^-6), median_SEPT9_cancer,
 type="o", pch=16, cex=0.5, col="red", xlab="", ylab="",
 main="SEPT9: Median Methylation by Group", ylim=c(0,1.1))
title(ylab="Methylation Level", line=2.2, cex.lab=1.2)
title(xlab="Position (Mb)", line=2.2, cex.lab=1.2)
points(rep(start(ranges(SEPT9_rel))*10^-6), median_SEPT9_control,
 pch=17, col="blue", cex=0.5, type="o")
abline(v=75378036*10^-6, col="black", lwd=1)
legend(x=(75378036*10^-6)-0.000015, y=1.16, legend="CpG-SNP:
 intronic in open sea", bty="n", cex=.8)
legend(x=(start(ranges(SEPT9_rel))[1]*10^-6)+0.00012,y=1.17, pch=16,
 col="red", legend="Cancer", cex=1, bty="n")
legend(x=(start(ranges(SEPT9_rel))[1]*10^-6)-0.00002, y=1.17,
 pch=17, col="blue", legend="Control", cex=1, bty="n")
dev.off()


# RAB11
median_RAB11_cancer <- rowMedians(methLevel(RAB11_rel[,
 which(RAB11_raw@colData$Disease=="A")]), na.rm=T)
median_RAB11_control <- rowMedians(methLevel(RAB11_rel[,
 which(RAB11_raw@colData$Disease=="DVA_control")]), na.rm=T)
png(file="/Users/ecazaly/Dropbox/Thesis/Chapter_5/Drafts/
 RAB11_cancer.png", pointsize=12, units="mm", width=200, height=250,
 res=400)
par(mfrow=c(2,1), mar=c(5,3.3,4,1)+0.1)
plot(rep(start(ranges(RAB11_rel))*10^-6,ncol(cancer)),
 methLevel(RAB11_rel[, which(RAB11_raw@colData$Disease=="A")]),
 type="p", pch=16, cex=0.5, col="red", xlab="", ylab="",
 main="RAB11: Individual Methylation Levels", ylim=c(0,1.1))
title(ylab="Methylation Level", line=2.2, cex.lab=1.2)
title(xlab="Position (Mb)", line=2.2, cex.lab=1.2)
points(rep(start(ranges(RAB11_rel))*10^-6, ncol(control)),
 methLevel(RAB11_rel[, which(RAB11_raw@colData
 $Disease=="DVA_control")]), pch=17, col="blue", cex=0.5)
abline(v=8464538*10^-6, col="black", lwd=1)
legend(x=(8464538*10^-6)-0.00023, y=1.16, legend="CpG-SNP: intronic
 in CpG shore", bty="n", cex=.8)
points(rep(start(ranges(RAB11_rel))*10^-6), median_RAB11_cancer,
 type="o", pch=16, cex=0.5, col="red", xlab="Genomic Position (Mb)",
 ylab="Methylation", main="Median Methylation at RAB11 by cancer
 status")
```

```
points(rep(start(ranges(RAB11_rel))*10^-6), median_RAB11_control,
 pch=17, col="blue", cex=0.5, type="o")
legend(x=(start(ranges(RAB11_rel))[1]*10^-6)+0.00008,y=1.17, pch=16,
 col="red", legend="Cancer", cex=1, bty="n")
legend(x=(start(ranges(RAB11_rel))[1]*10^-6)-0.00001, y=1.17,
 pch=17, col="blue", legend="Control", cex=1, bty="n")
plot(rep(start(ranges(RAB11_rel))*10^-6), median_RAB11_cancer,
 type="o", pch=16, cex=0.5, col="red", xlab="", ylab="",
 main="RAB11: Median Methylation by Group", ylim=c(0,1.1))
title(ylab="Methylation Level", line=2.2, cex.lab=1.2)
title(xlab="Position (Mb)", line=2.2, cex.lab=1.2)
points(rep(start(ranges(RAB11_rel))*10^-6), median_RAB11_control,
 pch=17, col="blue", cex=0.5, type="o")
abline(v=8464538*10^-6, col="black", lwd=1)
legend(x=(8464538*10^-6)-0.00023, y=1.16, legend="CpG-SNP: intronic
 in CpG shore", bty="n", cex=.8)
legend(x=(start(ranges(RAB11_rel))[1]*10^-6)+0.00008,y=1.17, pch=16,
 col="red", legend="Cancer", cex=1, bty="n")
legend(x=(start(ranges(RAB11_rel))[1]*10^-6)-0.00001, y=1.17,
 pch=17, col="blue", legend="Control", cex=1, bty="n")
dev.off()


# ITGB2
median_ITGB2_cancer <- rowMedians(methLevel(ITGB2_rel[,
 which(ITGB2_raw@colData$Disease=="A")]), na.rm=T)
median_ITGB2_control <- rowMedians(methLevel(ITGB2_rel[,
 which(ITGB2_raw@colData$Disease=="DVA_control")]), na.rm=T)
png(file="/Users/ecazaly/Dropbox/Thesis/Chapter_5/Drafts/
 ITGB2_cancer.png", pointsize=12, units="mm", width=200, height=250,
 res=400)
par(mfrow=c(2,1), mar=c(5,3.3,4,1)+0.1)
plot(rep(start(ranges(ITGB2_rel))*10^-6,ncol(cancer)),
 methLevel(ITGB2_rel[, which(ITGB2_raw@colData$Disease=="A")]),
 type="p", pch=16, cex=0.5, col="red", xlab="", ylab="",
 main="ITGB2: Individual Methylation Levels", ylim=c(0,1.1))
title(ylab="Methylation Level", line=2.2, cex.lab=1.2)
title(xlab="Position (Mb)", line=2.2, cex.lab=1.2)
points(rep(start(ranges(ITGB2_rel))*10^-6, ncol(control)),
 methLevel(ITGB2_rel[, which(ITGB2_raw@colData
 $Disease=="DVA_control")]), pch=17, col="blue", cex=0.5)
abline(v=46349496*10^-6, col="black", lwd=1)
legend(x=(46349496*10^-6)+0.000002, y=1.16, legend="CpG-SNP: UTR in
 CpG shore", bty="n", cex=.8)
points(rep(start(ranges(ITGB2_rel))*10^-6), median_ITGB2_cancer,
 type="o", pch=16, cex=0.5, col="red", xlab="Genomic Position (Mb)",
 ylab="Methylation", main="Median Methylation at ITGB2 by cancer
```

```
 status")
points(rep(start(ranges(ITGB2_rel))*10^-6), median_ITGB2_control,
 pch=17, col="blue", cex=0.5, type="o")
legend(x=(start(ranges(ITGB2_rel))[1]*10^-6)+0.00008,y=1.17, pch=16,
 col="red", legend="Cancer", cex=1, bty="n")
legend(x=(start(ranges(ITGB2_rel))[1]*10^-6)-0.000015, y=1.17,
 pch=17, col="blue", legend="Control", cex=1, bty="n")
plot(rep(start(ranges(ITGB2_rel))*10^-6), median_ITGB2_cancer,
 type="o", pch=16, cex=0.5, col="red", xlab="", ylab="",
 main="ITGB2: Median Methylation by Group", ylim=c(0,1.1))
title(ylab="Methylation Level", line=2.2, cex.lab=1.2)
title(xlab="Position (Mb)", line=2.2, cex.lab=1.2)
points(rep(start(ranges(ITGB2_rel))*10^-6), median_ITGB2_control,
 pch=17, col="blue", cex=0.5, type="o")
abline(v=46349496*10^-6, col="black", lwd=1)
legend(x=(46349496*10^-6)+0.0000002, y=1.16, legend="CpG-SNP: UTR in
 CpG shore", bty="n", cex=.8)
legend(x=(start(ranges(ITGB2_rel))[1]*10^-6)+0.00008,y=1.17, pch=16,
 col="red", legend="Cancer", cex=1, bty="n")
legend(x=(start(ranges(ITGB2_rel))[1]*10^-6)-0.000015, y=1.17,
 pch=17, col="blue", legend="Control", cex=1, bty="n")
dev.off()
```